目 录

第1章 预备知识 …… 1

1.1 微积分回顾 …… 1

1.1.1 极限和连续性 …… 1

1.1.2 微分函数 …… 3

1.1.3 积分 …… 4

1.1.4 级数 …… 5

1.1.5 多项式求值 …… 6

1.1.6 微积分回顾的练习题 …… 8

1.2 二进制数 …… 9

1.2.1 二进制数 …… 10

1.2.2 序列与级数 …… 11

1.2.3 二进制分数 …… 12

1.2.4 二进制移位 …… 14

1.2.5 科学计数法 …… 14

1.2.6 机器数 …… 14

1.2.7 计算机精度 …… 15

1.2.8 计算机浮点数 …… 16

1.2.9 二进制数的练习 …… 16

1.3 误差分析 …… 18

1.3.1 截断误差 …… 19

1.3.2 舍入误差 …… 19

1.3.3 舍去和舍入 …… 20

1.3.4 精度损失 …… 20

1.3.5 $O(h^n)$阶逼近 …… 22

1.3.6 序列的收敛阶 …… 24

1.3.7 误差传播 …… 24

1.3.8 数据的不确定性 …… 27

1.3.9 误差分析的练习 …… 27

1.3.10 算法和程序 …… 29

第2章 非线性方程 $f(x)=0$ 的解法 …… 30

2.1 求解 $x=g(x)$ 的迭代法 …… 30

2.1.1 寻求固定点 …… 31

2.1.2 固定点迭代的图形解释 …… 34

2.1.3 绝对误差和相对误差 …… 35

2.1.4 求解 $x=g(x)$ 迭代过程的练习 …… 36

2.1.5 算法和程序 …… 37
2.2 定位一个根的划分方法(bracketing methods) …… 37
2.2.1 波尔察诺(Bolzano)二分法 …… 39
2.2.2 试值法的收敛性 …… 42
2.2.3 划分方法练习 …… 45
2.2.4 算法和程序 …… 46
2.3 初始近似值和收敛判定准则 …… 46
2.3.1 检测收敛性 …… 47
2.3.2 有问题的函数(TroubleSome Functions) …… 49
2.3.3 初始近似值的练习 …… 50
2.3.4 算法和程序 …… 51
2.4 牛顿拉夫申(Newton - Raphson)法和割线法 …… 51
2.4.1 求根的斜率法 …… 51
2.4.2 被零除错误 …… 54
2.4.3 收敛速度 …… 55
2.4.4 缺陷 …… 57
2.4.5 割线法 …… 58
2.4.6 加速收敛 …… 60
2.4.7 牛顿拉夫申法和割线法的练习 …… 62
2.4.8 算法和程序 …… 64
2.5 Aitken 过程、Steffensen 法和 Muller 法(可选) …… 65
2.5.1 Aitken 过程 …… 66
2.5.2 Muller 法 …… 67
2.5.3 方法之间的比较 …… 68
2.5.4 Aitken 法、Steffensen 法和 Muller 法的练习 …… 72
2.5.5 算法和程序 …… 73
第3章 线性方程组 $AX = B$ 的数值解法 …… 74
3.1 向量和矩阵介绍 …… 74
3.1.1 矩阵和二维数组 …… 76
3.1.2 向量和矩阵简介的练习 …… 79
3.2 向量和矩阵的性质 …… 80
3.2.1 矩阵乘 …… 81
3.2.2 特殊矩阵 …… 82
3.2.3 非奇异矩阵的逆 …… 83
3.2.4 行列式 …… 83
3.2.5 平面旋转 …… 85
3.2.6 MATLAB …… 86
3.2.7 向量和矩阵性质的练习 …… 87
3.2.8 算法和程序 …… 88
3.3 上三角线性方程组 …… 89
3.3.1 上三角线性方程组的练习 …… 92

3.3.2 算法和程序 …… 92
3.4 高斯消去法和选主元 …… 93
3.4.1 选主元以避免 $a_{pp}^{(p)}=0$ …… 97
3.4.2 选主元以减少误差 …… 97
3.4.3 病态情况 …… 99
3.4.4 MATLAB …… 100
3.4.5 高斯消去法和选主元的练习 …… 102
3.4.6 算法和程序 …… 104
3.5 三角分解法 …… 105
3.5.1 线性方程组的解 …… 105
3.5.2 三角分解法 …… 106
3.5.3 计算复杂性 …… 110
3.5.4 置换矩阵 …… 110
3.5.5 扩展高斯消去过程 …… 112
3.5.6 MATLAB …… 112
3.5.7 三角分解法的练习 …… 114
3.5.8 算法和程序 …… 115
3.6 求解线性方程组的迭代法 …… 117
3.6.1 雅克比迭代 …… 117
3.6.2 Gauss-Seidel 迭代法 …… 119
3.6.3 收敛性 …… 121
3.6.4 求解线性方程组的迭代法的练习 …… 123
3.6.5 算法和程序 …… 124
3.7 非线性方程组的迭代法:Seidel 法和牛顿法(可选) …… 125
3.7.1 理论 …… 127
3.7.2 广义微分 …… 128
3.7.3 接近固定点处的收敛性 …… 129
3.7.4 Seidel 迭代 …… 130
3.7.5 求解非线性方程组的牛顿法 …… 131
3.7.6 牛顿法概要 …… 132
3.7.7 MATLAB …… 133
3.7.8 求解非线性方程组的迭代法的练习 …… 135
3.7.9 算法和程序 …… 138
第4章 插值与多项式逼近 …… 140
4.1 泰勒级数和函数计算 …… 141
4.1.1 多项式计算方法 …… 145
4.1.2 习题 …… 145
4.1.3 算法与程序 …… 148
4.2 插值介绍 …… 149
4.2.1 习题 …… 153
4.2.2 算法与程序 …… 154

4.3 拉格朗日逼近 …… 154
4.3.1 误差项和误差界 …… 158
4.3.2 比较精度与 $O(h^{N+1})$ …… 160
4.3.3 MATLAB …… 162
4.3.4 习题 …… 163
4.3.5 算法与程序 …… 164
4.4 牛顿多项式 …… 165
4.4.1 嵌套乘法 …… 166
4.4.2 多项式逼近、节点及中心 …… 166
4.4.3 习题 …… 170
4.4.4 算法与程序 …… 172
4.5 切比雪夫多项式(可选) …… 172
4.5.1 切比雪夫多项式性质 …… 173
4.5.2 最小上界 …… 174
4.5.3 等距节点 …… 175
4.5.4 切比雪夫节点 …… 175
4.5.5 龙格现象 …… 176
4.5.6 区间变换 …… 177
4.5.7 正交性质 …… 178
4.5.8 MATLAB …… 179
4.5.9 习题 …… 180
4.5.10 算法与程序 …… 181
4.6 帕德逼近 …… 182
4.6.1 连分式 …… 184
4.6.2 习题 …… 185
4.6.3 算法与程序 …… 186
第5章 曲线拟合 …… 188
5.1 最小二乘拟合曲线 …… 188
5.1.1 求最小二乘曲线 …… 189
5.1.2 幂函数拟合 $y=Ax^M$ …… 191
5.1.3 最小二乘拟合曲线的练习 …… 192
5.1.4 算法和程序 …… 195
5.2 曲线拟合 …… 196
5.2.1 对 $y=Ce^{Ax}$ 线性化方法 …… 196
5.2.2 求解 $y=Ce^{Ax}$ 的非线性最小二乘法 …… 197
5.2.3 数据线性化变换 …… 199
5.2.4 线性最小二乘法 …… 200
5.2.5 矩阵公式 …… 201
5.2.6 多项式拟合 …… 201
5.2.7 多项式摆动 …… 203
5.2.8 曲线拟合的练习 …… 204

5.2.9 算法和程序 …… 207
5.3 样条函数插值 …… 207
5.3.1 分段线性插值 …… 208
5.3.2 分段三次样条曲线 …… 209
5.3.3 三次样条的存在性 …… 209
5.3.4 构造三次样条 …… 210
5.3.5 端点约束 …… 212
5.3.6 三次样条曲线的适宜性 …… 216
5.3.7 样条函数插值的练习 …… 218
5.3.8 算法和程序 …… 220
5.4 傅里叶级数和三角多项式 …… 221
5.4.1 三角多项式逼近 …… 225
5.4.2 傅里叶级数和三角多项式的练习 …… 228
5.4.3 算法和程序 …… 229
第6章 数值微分 …… 230
6.1 导数的近似值 …… 230
6.1.1 差商的极限 …… 230
6.1.2 中心差分公式 …… 232
6.1.3 误差分析和优化步长 …… 234
6.1.4 Richardson 外推法 …… 237
6.1.5 导数近似值的练习 …… 240
6.1.6 算法和程序 …… 243
6.2 数值差分公式 …… 243
6.2.1 更多的中心差分公式 …… 243
6.2.2 误差分析 …… 245
6.2.3 拉格朗日多项式微分 …… 247
6.2.4 牛顿多项式微分 …… 249
6.2.5 数值微分公式的练习 …… 251
6.2.6 算法和程序 …… 253
第7章 数值积分 …… 254
7.1 积分简介 …… 255
7.1.1 习题 …… 261
7.2 组合梯形公式和辛普生公式 …… 263
7.2.1 误差分析 …… 265
7.2.2 习题 …… 270
7.2.3 算法与程序 …… 272
7.3 递归公式与龙贝格积分 …… 273
7.3.1 龙贝格积分 …… 277
7.3.2 习题 …… 282
7.3.3 算法与程序 …… 284
7.4 自适应积分 …… 284

7.4.1 区间细分(refinement) …… 285
7.4.2 精度测试 …… 285
7.4.3 算法与程序 …… 289
7.5 高斯 - 勒让德积分(可选) …… 290
7.5.1 习题 …… 294
7.5.2 算法与程序 …… 296
第8章 数值优化 …… 297
8.1 函数极小值 …… 297
8.1.1 搜索方法 …… 298
8.1.2 求解 $f(x,y)$ 的极值 …… 300
8.1.3 Nelder - Mead 法 …… 301
8.1.4 根据导数求极小值 …… 304
8.1.5 最速下降法 …… 306
8.1.6 求解函数极小值的练习 …… 315
8.1.7 算法和程序 …… 316
第9章 微分方程求解 …… 318
9.1 微分方程导论 …… 318
9.1.1 初值问题 …… 319
9.1.2 几何解释 …… 320
9.1.3 习题 …… 321
9.2 欧拉方法 …… 323
9.2.1 几何描述 …… 324
9.2.2 步长与误差 …… 325
9.2.3 习题 …… 327
9.2.4 算法与程序 …… 328
9.3 休恩方法 …… 330
9.3.1 步长与误差 …… 330
9.3.2 习题 …… 333
9.3.3 算法与程序 …… 334
9.4 泰勒级数法 …… 335
9.4.1 习题 …… 339
9.4.2 算法与程序 …… 340
9.5 龙格 - 库塔方法 …… 340
9.5.1 关于该方法的讨论 …… 342
9.5.2 步长与误差 …… 342
9.5.3 $N=2$ 的龙格 - 库塔方法 …… 345
9.5.4 龙格 - 库塔 - 费尔博格方法(RKF45) …… 346
9.5.5 习题 …… 350
9.5.6 算法与程序 …… 351
9.6 预测 - 校正方法 …… 353
9.6.1 阿达姆斯 - 巴什弗斯 - 摩尔顿方法 …… 353

9.6.2 误差估计与校正 …… 354
9.6.3 实际考虑 …… 354
9.6.4 米尔尼－辛普生方法 …… 355
9.6.5 误差估计与校正 …… 355
9.6.6 正确的步长 …… 357
9.6.7 习题 …… 361
9.6.8 程序与算法 …… 362
9.7 微分方程组 …… 363
9.7.1 数值解 …… 363
9.7.2 高阶微分方程 …… 365
9.7.3 习题 …… 366
9.7.4 算法与程序 …… 368
9.8 边值问题 …… 370
9.8.1 分解为两个初值问题:线性打靶法 …… 371
9.8.2 习题 …… 375
9.8.3 算法与程序 …… 376
9.9 有限差分方法 …… 376
9.9.1 习题 …… 382
9.9.2 算法与程序 …… 382
第 10 章 偏微分方程数值解 …… 384
10.1 双曲型方程 …… 385
10.1.1 波动方程 …… 385
10.1.2 差分方程 …… 386
10.1.3 初始值 …… 387
10.1.4 D'Alembert 方法 …… 387
10.1.5 给定的两个确定行 …… 388
10.1.6 双曲线型方程的练习 …… 391
10.1.7 算法和程序 …… 392
10.2 抛物型方程 …… 393
10.2.1 热传导方程 …… 393
10.2.2 差分方程 …… 393
10.2.3 Crank－Nicholson 法 …… 396
10.2.4 抛物型方程的练习 …… 400
10.2.5 算法和程序 …… 401
10.3 椭圆型方程 …… 402
10.3.1 Laplace 差分方程 …… 402
10.3.2 建立线性方程组 …… 403
10.3.3 导数边界条件 …… 405
10.3.4 迭代方法 …… 407
10.3.5 Poisson 方程和 Helmholtz 方程 …… 410
10.3.6 改进 …… 410

10.3.7 椭圆型方程的练习 …… 411
10.3.8 算法和程序 …… 413
第 11 章 特征值与特征向量 …… 415
11.1 齐次方程组:特征值问题 …… 415
11.1.1 背景知识 …… 415
11.1.2 特征值 …… 417
11.1.3 对角化 …… 420
11.1.4 对称性的优势 …… 422
11.1.5 特征值范围估计 …… 423
11.1.6 方法综述 …… 423
11.1.7 齐次方程组:特征值问题的练习 …… 423
11.2 幂方法 …… 424
11.2.1 收敛速度 …… 428
11.2.2 移位反幂法 …… 428
11.2.3 幂法的练习 …… 432
11.2.4 算法和程序 …… 433
11.3 雅克比方法 …… 434
11.3.1 平面旋转变换 …… 434
11.3.2 相似和正交变换 …… 435
11.3.3 雅克比序列的变换 …… 436
11.3.4 一般步骤 …… 436
11.3.5 使 d_{pq} 和 d_{qp} 为零 …… 437
11.3.6 一般步骤总结 …… 438
11.3.7 修正矩阵的特征值 …… 439
11.3.8 消去 a_{pq} 的策略 …… 439
11.3.9 雅克比法的练习 …… 442
11.3.10 算法和程序 …… 443
11.4 对称矩阵的特征值 …… 444
11.4.1 Householder 法 …… 444
11.4.2 Householder 变换 …… 446
11.4.3 三对角形式归约 …… 448
11.4.4 *QR* 法 …… 450
11.4.5 加速移位 …… 451
附录 MATLAB 介绍 …… 456
参考文献 …… 463
习题答案 …… 473

第1章　预备知识

假设函数 $f(x)=\cos(x)$，则它的导数 $f'(x)=-\sin(x)$，不定积分为 $F(x)=\sin(x)+C$。在微积分学中可以学到这些公式，前者确定函数曲线 $y=f(x)$ 在点 $(x_0, f(x_0))$ 的斜率 $m=f'(x_0)$；后者可计算出函数曲线在 $a \leqslant x \leqslant b$ 范围下的面积。

曲线 y 在点 $(\pi/2, 0)$ 的斜率 $m=f'(\pi/2)=-1$，可通过它找到在这一点的切线（如图1.1(a)所示）：

$$y_{\tan}=m\left(x-\frac{\pi}{2}\right)+0=f'\left(\frac{\pi}{2}\right)\left(x-\frac{\pi}{2}\right)=-x+\frac{\pi}{2}$$

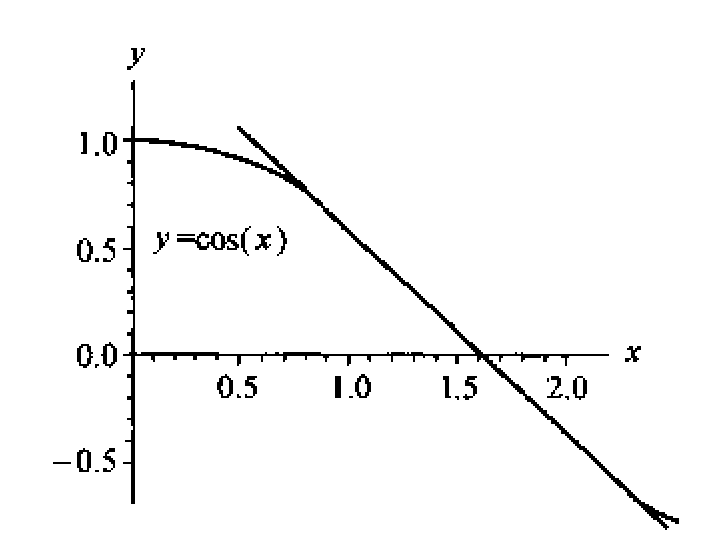


图1.1(a)　函数曲线 $y=\cos(x)$ 在点 $(\pi/2,0)$ 的切线

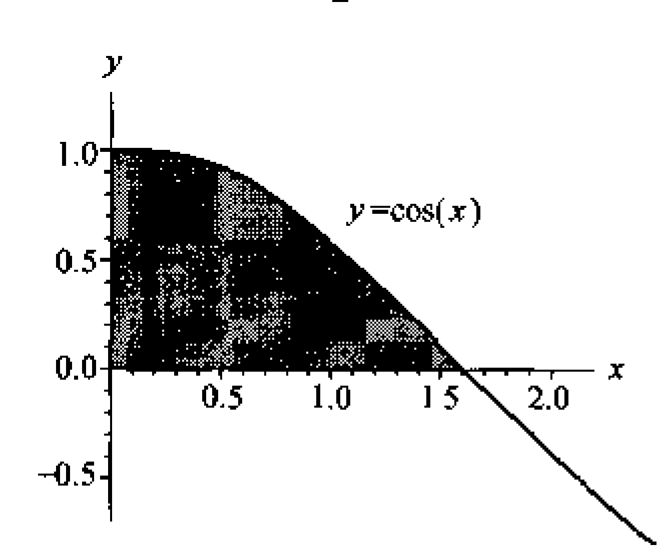


图1.1(b)　函数曲线 $y=\cos(x)$ 在区间 $[0,\pi/2]$ 下的区域

通过积分方法计算曲线 y 在 $0 \leqslant x \leqslant \pi/2$ 范围下的面积为（如图1.1(b)所示）：

$$\text{面积}=\int_0^{\pi/2}\cos(x)\,dx=F\left(\frac{\pi}{2}\right)-F(0)=\sin\left(\frac{\pi}{2}\right)-0=1$$

1.1　微积分回顾

本书假定读者具有大学本科的微积分知识，熟悉极限、连续性、求导、积分、序列和级数等。其中，本书中将用到的微积分知识如下：

1.1.1　极限和连续性

定义1.1　设 $f(x)$ 为定义在实数集合 S 上的函数，如果对于任意给定的 $\varepsilon>0$，总存在 $\delta>0$，使得对于任意 $x \in S$，且 $0<|x-x_0|<\delta$，有 $|f(x)-L|<\varepsilon$。则称函数 f 在 $x=x_0$ 处的极限为 L，表示为：

$$\lim_{x \to x_0} f(x)=L \tag{1}$$

当采用 h 增量表达式 $x=x_0+h$ 时，式(1)可表示为：

$$\lim_{h\to 0} f(x_0+h)=L \tag{2}$$

定义 1.2 设 $f(x)$为定义在实数集合 S 上的函数,且 $x_0\in S$,如果:

$$\lim_{x\to x_0} f(x)=f(x_0) \tag{3}$$

则函数 f 在点 $x=x_0$ 处连续。

如果函数 f 在任意点 $x\in S$ 上连续,则函数 f 在集合 S 上连续。表达式 $C^n(S)$表示函数 f 自身和它的前 n 阶导数在集合 S 上连续的所有函数f 的集合。当 S 为区间$[a,b]$时,则可用表达式 $C^n[a,b]$来表示。例如函数$f(x)=x^{4/3}$,其定义域为$[1,-1]$,则显然 $f(x)$和$f'(x)=(4/3)x^{1/3}$在区间$[1,-1]$上连续,但$f''(x)=(4/9)x^{-2/3}$在 $x=0$ 处不连续。

定义 1.3 设$\{x_n\}_{n=1}^{\infty}$为一无限序列,如果对于给定的任意小的正数 $\epsilon>0$,总存在一个正整数 $N=N(\epsilon)$,使得当 $n>N$ 时,有$|x_n-L|<\epsilon$,则称序列$\{x_n\}_{n=1}^{\infty}$有极限 L,并记作:

$$\lim_{n\to\infty} x_n=L \tag{4}$$

当序列有极限时,则称其为收敛序列。另一个常用的表示形式为"当 $n\to\infty$ $x_n\to L$ 时"。式(4)等价于:

$$\lim_{n\to\infty}(x_n-L)=0 \tag{5}$$

这样,可将序列$\{\epsilon_n\}_{n=1}^{\infty}=\{x_n-L\}_{n=1}^{\infty}$看作一个误差序列。下列定理与连续性和收敛序列有关。

定理 1.1 设 $f(x)$为定义在实数集合 S 上的函数,且 $x_0\in S$,则下列命题是等价的:

(a) 函数 f 在 x_0 处连续

(b) 如果 $\lim_{n\to\infty} x_n=x_0$,则 $\lim_{n\to\infty} f(x_n)=f(x_0)$ (6)

定理 1.2(中值定理) 若 $f\in C[a,b]$,且 L 为$f(a)$与 $f(b)$之间的任意值,则存在 $c\in(a,b)$,有 $f(c)=L$。

例 1.1 函数$f(x)=\cos(x-1)$在区间$[0,1]$内连续,且常量 $L=0.8\in(\cos(0),\cos(1))$。函数 $f(x)=0.8$ 在区间$[0,1]$的解为 $c_1=0.356499$。同样,函数 $f(x)$在区间$[1,2.5]$内连续,且 $L=0.8\in(\cos(2.5),\cos(1))$。函数 $f(x)=0.8$ 在区间$[1,2.5]$的解 $c_2=1.643502$。这两种情况如图 1.2 所示。

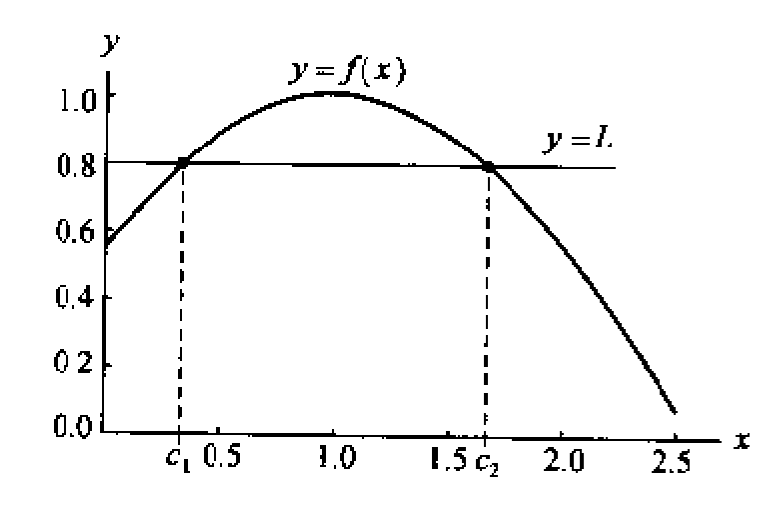


图 1.2 将中值定理应用在函数 $f(x)=\cos(x-1)$ 上,区间分别为$[0,1]$和$[1,2.5]$

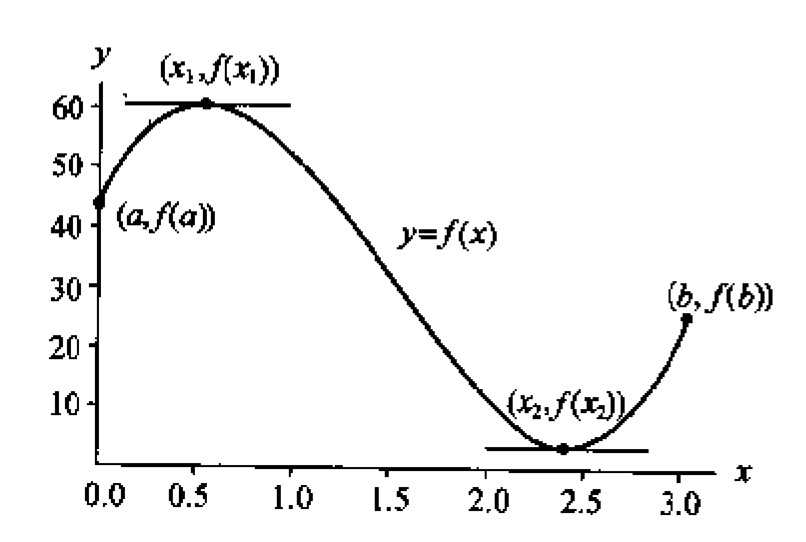


图 1.3 将极值定理应用在函数 $f(x)=35+59.5x-66.5x^2+15x^3$ 上,区间为$[0,3]$

定理 1.3(连续函数的极值定理) 设 $f \in C[a,b]$，则存在下界 M_1 和上界 M_2，以及 x_1，$x_2 \in [a,b]$，满足：

$$M_1 = f(x_1) \leqslant f(x) \leqslant f(x_2) = M_2 \qquad x \in [a,b] \tag{7}$$

有时也可表示为：

$$M_1 = f(x_1) = \min_{a \leqslant x \leqslant b}\{f(x)\} \text{和} M_2 = f(x_2) = \max_{a \leqslant x \leqslant b}\{f(x)\} \tag{8}$$

1.1.2 微分函数

定义 1.4 设 $f(x)$在一个包含 x_0 的开区间内有定义。如果：

$$\lim_{x \to x_0} \frac{f(x) - f(x_0)}{x - x_0} \tag{9}$$

存在，则称函数 f 在点 x_0 处可微。如果此极限存在，则可表示为$f'(x_0)$，并称之为 f 在点 x_0 处的导数。也可以采用 h 增量表达式来表示此极限：

$$\lim_{h \to 0} \frac{f(x_0 + h) - f(x_0)}{h} = f'(x_0) \tag{10}$$

如果函数在集合 S 上的每一点都存在导数，则称函数在集合 S 上可微。要注意的是，数 $m = f'(x_0)$是函数曲线 $f(x)$在点$(x_0, f(x_0))$的切线斜率。

定理 1.4 如果函数 $f(x)$在点 $x = x_0$ 处可微，则 $f(x)$在点 $x = x_0$ 处连续。

根据定理 1.3，如果函数 f 在闭区间$[a,b]$内可微，则函数 f 的极值在闭区间的端点或在开区间(a,b)的临界点(即$f'(x) = 0$ 的解)。

例 1.2 函数 $f(x) = 15x^3 - 66.5x^2 + 59.5x + 35$ 在区间$[0,3]$内可微。$f'(x) = 45x^2 - 123x + 59.5 = 0$ 的解为 $x_1 = 0.54955$ 和 $x_2 = 2.40601$。函数 f 在区间$[0,3]$内的极小值和极大值分别为：

$$\min\{f(0), f(3), f(x_1), f(x_2)\} = \min\{35, 20, 50.10438, 2.11850\} = 2.11850$$

和

$$\max\{f(0), f(3), f(x_1), f(x_2)\} = \max\{35, 20, 50.10438, 2.11850\} = 50.10438$$

定理 1.5(罗尔定理) 设 $f \in C[a,b]$，且对于所有 $x \in (a,b)$，存在导数$f'(x)$，如果$f(a) = f(b) = 0$，则至少存在一点 $c \in (a,b)$满足$f'(c) = 0$。

定理 1.6(均值定理或微分中值定理) 若 $f \in C[a,b]$，且对于所有 $x \in (a,b)$，存在导数 $f'(x)$，则至少存在一点 $c \in (a,b)$，使得：

$$f'(c) = \frac{f(b) - f(a)}{b - a} \tag{11}$$

成立。

均值定理的几何意义是：函数曲线 $y = f(x)$至少存在一点$(c, f(c))$，该点上切线的斜率等于过点$(a, f(a))$和点$(b, f(b))$的割线的斜率。

例 1.3 函数 $f(x) = \sin(x)$在闭区间$[0.1, 2.1]$内连续，且在开区间$(0.1, 2.1)$内可微，则根据均值定理，至少存在 c，满足：

$$f'(c)=\frac{f(2.1)-f(0.1)}{2.1-0.1}=\frac{0.863209-0.099833}{2.1-0.1}=0.381688$$

$f'(c)=\cos(c)=0.381688$ 在区间(0.1,2.1)上的解为 $c=1.179174$。函数曲线 $f(x)$、割线 $y=0.381688x+0.099833$ 和切线 $y=0.381688x+0.474215$ 如图 1.4 所示。

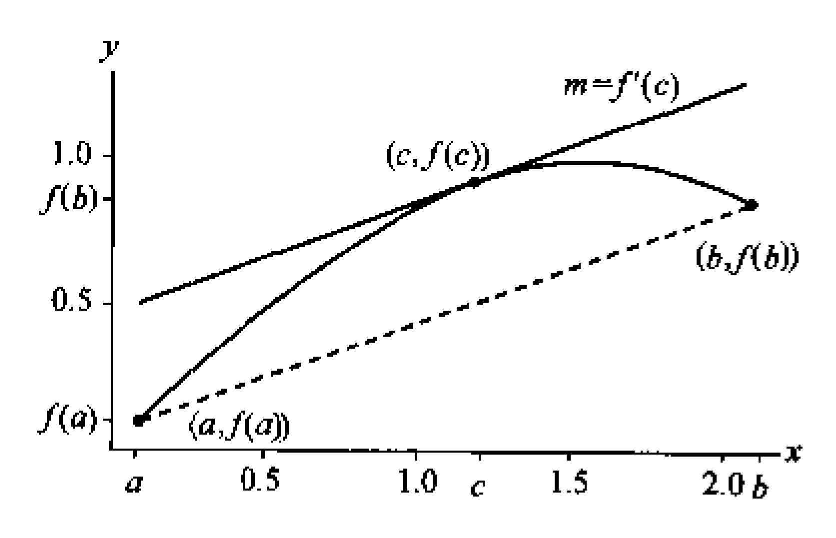


图 1.4 将均值定理运用于函数 $f(x)=\sin(x)$ 上,区间为[0.1,2.1]

定理 1.7(广义罗尔定理) 若 $f\in C[a,b]$,$f'(x),f''(x),\cdots,f^{(n)}(x)$ 在开区间 (a,b) 内存在,且 $x_0,x_1,\cdots,x_n\in[a,b]$,如果对于 $f(x_j)=0$ 有 $j=0,1,\cdots,n$,则至少存在一点 $c\in(a,b)$,满足 $f^{(n)}(c)=0$。

1.1.3 积分

定理 1.8(第一基本定理) 如果函数 f 在区间 $[a,b]$ 内连续,且函数 F 是 f 在区间 $[a,b]$ 内的任一原函数,则:

$$\int_a^b f(x)dx = F(b) - F(a),\text{其中 } F'(x) = f(x) \tag{12}$$

定理 1.9(第二基本定理) 如果函数 f 在区间 $[a,b]$ 内连续,且 $x\in(a,b)$,则:

$$\frac{d}{dx}\int_a^x f(t)dt = f(x) \tag{13}$$

例 1.4 函数 $f(x)=\cos(x)$ 在区间 $[0,\pi/2]$ 内满足定理 1.9 的假设,则根据定理的结论可得:

$$\frac{d}{dx}\int_0^{x^2}\cos(t)dt = \cos(x^2)(x^2)' = 2x\cos(x^2)$$

定理 1.10(积分均值定理) 若 $f\in C[a,b]$,则至少存在一点 $c\in(a,b)$ 满足:

$$\frac{1}{b-a}\int_a^b f(x)dx = f(c)$$

$f(c)$ 是函数 f 在区间 $[a,b]$ 内的平均值。

例 1.5 函数 $f(x)=\sin(x)+\frac{1}{3}\sin(3x)$ 在区间 $[0,2.5]$ 内满足定理 1.10 的假设。函数 $f(x)$ 的一个原函数为 $F(x)=-\cos(x)-\frac{1}{9}\cos(3x)$,则函数 $f(x)$ 在区间 $[0,2.5]$ 内的平均值为:

$$\frac{1}{2.5-0}\int_0^{2.5}f(x)dx = \frac{F(2.5)-F(0)}{2.5} = \frac{0.762629-(-1.111111)}{2.5}$$

$$= \frac{1.873740}{2.5} = 0.749496$$

方程 $f(c) = 0.749496$ 在区间[0,2.5]内有 3 个解：$c_1 = 0.440566$、$c_2 = 1.268010$ 和 $c_3 = 1.873583$。长为 $b - a = 2.5$，高为 $f(c_j) = 0.749496$ 的长方形区域面积为 $f(c_j)(b-a) = 1.873740$。此长方形区域的面积值与函数 $f(x)$ 在区间[0,2.5]内的积分值相同，二者的比较如图 1.5 所示。

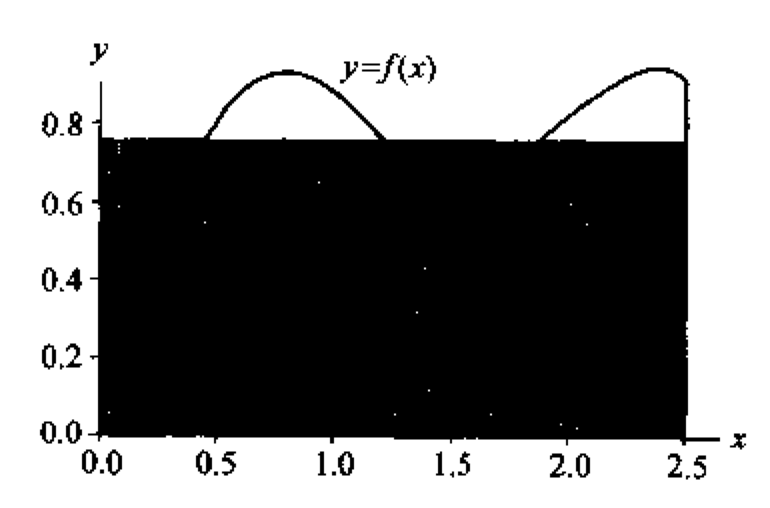


图 1.5 将积分均值定理运用在 $f(x) = \sin(x) + \frac{1}{3}\sin(3x)$ 上，区间为[0,2.5]

定理 1.11（加权积分均值定理） 若 $f, g \in C[a,b]$，且当 $g(x) \geq 0$ 时有 $x \in [a,b]$，则至少存在一点 $c \in (a,b)$，满足：

$$\int_a^b f(x) g(x) dx = f(c) \int_a^b g(x) dx \tag{14}$$

例 1.6 函数 $f(x) = \sin(x)$ 和函数 $g(x) = x^2$ 在区间[0,π/2]内满足定理 1.11 的假设，则至少存在一点 c，满足：

$$\sin(c) = \frac{\int_0^{\pi/2} x^2 \sin(x) dx}{\int_0^{\pi/2} x^2 dx} = \frac{1.14159}{1.29193} = 0.883631$$

或 $c = \arcsin(0.883631) = 1.08356$

1.1.4 级数

定义 1.5 设有序列 $\{a_n\}_{n=1}^{\infty}$，则 $\sum_{n=1}^{\infty} a_n$ 为一无穷级数，且第 n 个部分和为 $S_n = \sum_{k=1}^{n} a_k$。当且仅当序列 $\{S_n\}_{n=1}^{\infty}$ 收敛于极限 S 时，无穷级数收敛，可表示为：

$$\lim_{n\to\infty} S_n = \lim_{n\to\infty} \sum_{k=1}^{n} a_k = S \tag{15}$$

如果级数不收敛，则称为级数发散。

例 1.7 已知无穷序列 $\{a_n\}_{n=1}^{\infty} = \left\{\frac{1}{n(n+1)}\right\}_{n=1}^{\infty}$，则第 n 个部分和为：

$$S_n = \sum_{k=1}^{n} \frac{1}{k(k+1)} = \sum_{k=1}^{n} \left(\frac{1}{k} - \frac{1}{k+1}\right) = 1 - \frac{1}{n+1}$$

因此，无穷级数的和为：

$$S=\lim_{n\to\infty}S_n=\lim_{n\to\infty}\left(1-\frac{1}{n+1}\right)=1$$

定理 1.12(泰勒定理) 若函数 $f\in C^{n+1}[a,b]$,且 $x_0\in[a,b]$,则对任意 $x\in(a,b)$,都存在 $c=c(x)$(c 依赖于 x)位于 x_0 和 x 之间,满足:

$$f(x)=P_n(x)+R_n(x) \tag{16}$$

其中:

$$P_n(x)=\sum_{k=0}^{n}\frac{f^{(k)}(x_0)}{k!}(x-x_0)^k \tag{17}$$

而且:

$$R_n(x)=\frac{f^{(n+1)}(c)}{(n+1)!}(x-x_0)^{n+1} \tag{18}$$

例 1.8 函数 $f(x)=\sin(x)$满足定理 1.12 的假设,则通过将函数 $f(x)$在 $x=0$ 处的各阶导数值带入式(17),可得到在 $x_0=0$ 处 $n=9$ 的 n 阶泰勒多项式 $P_n(x)$:

$$\begin{aligned}
f(x)&=\sin(x), & f(0)&=0,\\
f'(x)&=\cos(x), & f'(0)&=1,\\
f''(x)&=-\sin(x), & f''(0)&=0,\\
f^{(3)}(x)&=-\cos(x), & f^{(3)}(0)&=-1,\\
&\vdots & &\vdots\\
f^{(9)}(x)&=\cos(x), & f^{(9)}(0)&=1,
\end{aligned}$$

$$P_9(x)=x-\frac{x^3}{3!}+\frac{x^5}{5!}-\frac{x^7}{7!}+\frac{x^9}{9!}$$

函数 f 和 P_9 在区间$[0,2\pi]$内的图形表示如图 1.6 所示。

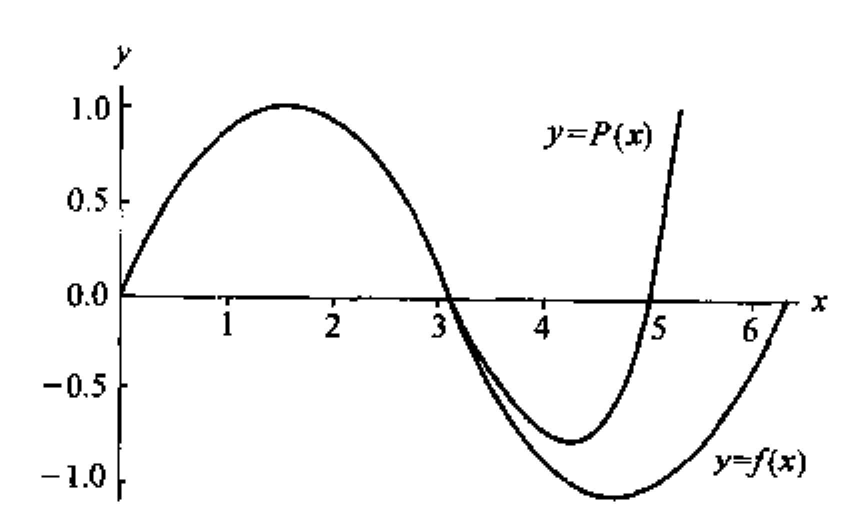


图 1.6 函数 $f(x)=\sin(x)$和泰勒多项式 $P(x)=x-x^3/3!+x^5/5!-x^7/7!+x^9/9!$ 的图形表示

推论 1.1 如果 $P_n(x)$是定理 1.12 中的 n 阶泰勒多项式,则:

$$P_n^{(k)}(x_0)=f^{(k)}(x_0),\ k=0,1,\cdots,n \tag{19}$$

1.1.5 多项式求值

设 n 阶多项式有如下形式:

$$P(x)=a_nx^n+a_{n-1}x^{n-1}+\cdots+a_2x^2+a_1x+a_0 \tag{20}$$

霍纳(Horner)方法[①] 或综合除法是计算多项式的一种技术,它可看作是一种嵌套乘法。例如,一个5阶多项式可改写为嵌套乘法的形式:

$$P_5(x)=((((a_5x+a_4)x+a_3)x+a_2)x+a_1)x+a_0$$

定理1.13(用于多项式计算的霍纳方法) 若有如式(20)的多项式,且 $x=c$ 是用于计算 $P(c)$ 的数。

设 $b_n=a_n$,并计算:

$$b_k=a_k+cb_{k+1}, \text{其中 } k=n-1,n-2,\cdots,1,0 \tag{21}$$

则 $b_0=P(c)$。进一步考虑,如果:

$$Q_0(x)=b_nx^{n-1}+b_{n-1}x^{n-2}+\cdots+b_3x^2+b_2x+b_1 \tag{22}$$

则:

$$P(x)=(x-c)Q_0(x)+R_0 \tag{23}$$

其中 $Q_0(x)$是 $n-1$ 阶多项式商,$R_0=b_0=P(c)$是余数。

证明:在式(23)中,用式(22)的右边替换 $Q_0(x)$,用 b_0 替换 R_0,则可得到:

$$\begin{aligned}P(x)&=(x-c)(b_nx^{n-1}+b_{n-1}x^{n-2}+\cdots+b_3x^2+b_2x+b_1)+b_0\\&=b_nx^n+(b_{n-1}-cb_n)x^{n-1}+\cdots+(b_2-cb_3)x^2\\&\quad+(b_1-cb_2)x+(b_0-cb_1)\end{aligned} \tag{24}$$

通过比较式(20)和式(24)的 x^k 的系数可确定 b_k 的值,如表1.1所示。

将 $x=c$ 替换入公式(22),且由于 $R_0=b_0$,可容易得出结论 $P(c)=b_0$,如下:

$$P(c)=(c-c)Q_0(c)+R_0=b_0 \tag{25}$$

证毕。

借助于计算机,可以很容易地实现式(21)中计算 b_k 的递归公式。一个简单的算法如下所示:

```
b(n) = a(n);
for k = n - 1: - 1:0
  b(k) = a(k) + c * b(k + 1);
end
```

表1.1 用于霍纳方法的系数 b_k

x^k	比较式(20)和式(24)	求解 b_k
x^n	$a_n=b_n$	$b_n=a_n$
x^{n-1}	$a_{n-1}=b_{n-1}-cb_n$	$b_{n-1}=a_{n-1}+cb_n$
$\vdots$	$\vdots$	$\vdots$
x^k	$a_k=b_k-cb_{k+1}$	$b_k=a_k+cb_{k+1}$
$\vdots$	$\vdots$	$\vdots$
x^0	$a_0=b_0-cb_1$	$b_0=a_0+cb_1$

当手工计算霍纳方法时,将 $P(x)$的系数写在一行,在 a_k 下计算 $b_k=a_k+cb_{k+1}$,这样更加方便。这一处理过程如表1.2所示。

---

① 秦九韶于1247年提出此方法,而霍纳于1819年提出同样的方法——译者注。

表 1.2 用于综合除法过程的霍纳表

输入	a_n	a_{n-1}	a_{n-2}	…	a_k	…	a_2	a_1	a_0
x		xb_n	xb_{n-1}	…	xb_{k+1}	…	xb_3	xb_2	xb_1
	b_n	b_{n-1}	b_{n-2}	…	b_k	…	b_2	b_1	$b_0 = P(x)$ 输出

例 1.9 利用综合除法(霍纳方法)求 $P(3)$,其中多项式为:

$$P(x)=x^5-6x^4+8x^3+8x^2+4x-40$$

	a_5	a_4	a_3	a_2	a_1	a_0
输入	1	-6	8	8	4	-40
$x=3$		3	-9	-3	15	57
	1	-3	-1	5	19	$17=P(3)=b_0$
	b_5	b_4	b_3	b_2	b_1	输出

因此,$P(3)=17$。

1.1.6 微积分回顾的练习题

1. (a) 求解 $L=\lim_{n\to\infty}(4n+1)/(2n+1)$,确定 $\{\epsilon_n\}=\{L-x_n\}$ 并求解 $\lim_{n\to\infty}\epsilon_n$。
   (b) 求解 $L=\lim_{n\to\infty}(2n^2+6n-1)/(4n^2+2n+1)$,确定 $\{\epsilon_n\}=\{L-x_n\}$ 并求解 $\lim_{n\to\infty}\epsilon_n$。
2. 设 $\{x_n\}_{n=1}^{\infty}$ 是满足 $\lim_{n\to\infty}x_n=2$ 的序列。
   (a) 求解 $\lim_{n\to\infty}\sin(x_n)$
   (b) 求解 $\lim_{n\to\infty}\ln(x_n^2)$
3. 根据中值定理,求解下列函数在指定区间内满足值 L 的数 c。
   (a) $f(x)=-x^2+2x+3$,区间为[-1,0],$L=2$
   (b) $f(x)=\sqrt{x^2-5x-2}$,区间为[6,8],$L=3$
4. 根据极值定理,求解下列函数在指定区间内的上界和下界。
   (a) $f(x)=x^2-3x+1$,区间为[-1,2]
   (b) $f(x)=\cos^2(x)-\sin(x)$,区间为$[0,2\pi]$
5. 根据罗尔定理,求解下列函数在指定区间内的 c 值。
   (a) $f(x)=x^4-4x^2$,区间为[-2,2]
   (b) $f(x)=\sin(x)+\sin(2x)$,区间为$[0,2\pi]$
6. 根据均值定理,求解下列函数在指定区间内的 c 值。
   (a) $f(x)=\sqrt{x}$,区间为[0,4]
   (b) $f(x)=\dfrac{x^2}{x+1}$,区间为[0,1]
7. 给定区间[0,3],将广义罗尔定理应用于函数 $f(x)=x(x-1)(x-3)$ 上。
8. 将第一基本定理应用于下列指定区间内的函数。

(a) $f(x)=xe^x$,区间为[0,2]

(b) $f(x)=\dfrac{3x}{x^2+1}$,区间为[-1,1]

9. 将第二基本定理应用于下列函数。

(a) $\dfrac{d}{dx}\int_0^x t^2\cos(t)\,dt$　　(b) $\dfrac{d}{dx}\int_1^{x^3} e^{t^2}\,dt$

10. 根据积分均值定理,求解下列函数在指定区间内的 c 值。

(a) $f(x)=6x^2$,区间为[-3,4]

(b) $f(x)=x\cos(x)$,区间为[0,3π/2]

11. 求解下列序列或级数的和。

(a) $\left\{\dfrac{1}{2^n}\right\}_{n=0}^{\infty}$　　(b) $\left\{\dfrac{2}{3^n}\right\}_{n=1}^{\infty}$

(c) $\sum_{n=1}^{\infty}\dfrac{3}{n(n+1)}$　　(d) $\sum_{k=1}^{\infty}\dfrac{1}{4k^2-1}$

12. 求解下列函数在 x_0 处的4阶泰勒多项式。

(a) $f(x)=\sqrt{x}$, $x_0=1$

(b) $f(x)=x^5+4x^2+3x+1$, $x_0=0$

(c) $f(x)=\cos(x)$, $x_0=0$

13. 设 $f(x)=\sin(x)$,且 $P(x)=x-x^3/3!+x^5/5!-x^7/7!+x^9/9!$。证明 $P^{(k)}(0)=f^{(k)}(0)$,其中 $k=1,2,\cdots,9$。

14. 利用综合除法(霍纳方法)求解 $P(c)$。

(a) $P(x)=x^4+x^3-13x^2-x-12$, $c=3$

(b) $P(x)=2x^7+x^6+x^5-2x^4-x+23$, $c=-1$

15. 求解中心位于原点,半径在1~3之间的所有圆的平均面积。

16. 设多项式 $P(x)$在区间[a,b]内有 n 个实根,证明 $P^{(n-1)}(x)$在区间[a,b]内至少有1个实根。

17. 设 f,f'和f''在区间[a,b]内有定义,且 $f(a)=f(b)=0$,当 $f(c)>0$, $c\in(a,b)$时,证明存在数 $d\in(a,b)$ 满足 $f''(d)<0$。

1.2 二进制数

在日常生活中,人们通常使用十进制数进行计算,但大多数计算机内部通常采用二进制数。虽然从表面上看,计算机的通信(输入、输出操作)是基于十进制数的,但这并不表示计算机内部采用的是十进制数。事实上,计算机首先将输入的十进制数转换成二进制数,然后进行二进制运算,最后将答案再转换成十进制数显示出来。可以通过下述实验对此进行验证。例如某计算机的最高精度为9位十进制数,它计算下述公式的结果为:

$$\sum_{k=1}^{100\,000} 0.1 = 9999.99447 \tag{1}$$

式(1)是对数$\frac{1}{10}$进行100 000次累加,而它的算术精确值应当是10 000。通过上述分析,说明计

算机的计算过程存在着明显的误差。在本节的最后,将具体描述当计算机将十进制小数$\frac{1}{10}$转换成二进制数时,误差是如何产生的。

1.2.1 二进制数

大多数的数学计算都采用十进制数。例如,十进制数1563的展开式为:

$$1563=(1\times10^3)+(5\times10^2)+(6\times10^1)+(3\times10^0)$$

通常,若N为一正整数,则存在数$a_0,a_1,\cdots,a_k$使得N的十进制扩展表达式为:

$$N=(a_k\times10^k)+(a_{k-1}\times10^{k-1})+\cdots+(a_1\times10^1)+(a_0\times10^0)$$

其中数a_k的取值范围为$\{0,1,\cdots,8,9\}$。则N的十进制表示为:

$$N=a_ka_{k-1}\cdots a_2a_1a_{0_{\text{ten}}}(\text{十进制}) \tag{2}$$

如果已知数为十进制,则式(2)可表示为:

$$N=a_ka_{k-1}\cdots a_2a_1a_0$$

例如,$1563=1563_{\text{ten}}$。

若用2的幂表示,则十进制数1563可表示为:

$$\begin{aligned}1563=&(1\times2^{10})+(1\times2^9)+(0\times2^8)+(0\times2^7)+(0\times2^6)\\&+(0\times2^5)+(1\times2^4)+(1\times2^3)+(0\times2^2)+(1\times2^1)\\&+(1\times2^0)\end{aligned} \tag{3}$$

可通过如下计算进行验证:

$$1563=1024+512+16+8+2+1$$

通常若N为一正整数,则存在数$b_0,b_1,\cdots,b_J$使得N的二进制展开表达式为:

$$N=(b_J\times2^J)+(b_{J-1}\times2^{J-1})+\cdots+(b_1\times2^1)+(b_0\times2^0) \tag{4}$$

其中数b_J为0或1。则N的二进制表示为:

$$N=b_Jb_{J-1}\cdots b_2b_1b_{0_{\text{two}}}(\text{二进制}) \tag{5}$$

利用式(5),可以将式(3)表示为:

$$1563=11000011011_{\text{two}}$$

注:单词"two"作为下标出现在二进制数的末尾,以便于读者区分十进制数和二进制数。因此,111表示数一百一十一,而111_{two}表示数七。

通常,由于2的幂的增长小于10的幂的增长,因此数N的二进制表示位数远大于它的十进制表示位数。

通过式(4)可得到一个计算N的二进制表示的有效算法。将式(4)的两边同除以2可得:

$$\frac{N}{2}=(b_J\times2^{J-1})+(b_{J-1}\times2^{J-2})+\cdots+(b_1\times2^0)+\frac{b_0}{2} \tag{6}$$

这样N除以2的余数是b_0。下面来计算b_1。如果式(6)表示为$N/2=Q_0+b_0/2$,则有:

$$Q_0=(b_J\times2^{J-1})+(b_{J-1}\times2^{J-2})+\cdots+(b_2\times2^1)+(b_1\times2^0) \tag{7}$$

将式(7)的两边同除以2可得:

$$\frac{Q_0}{2}=(b_J\times2^{J-2})+(b_{J-1}\times2^{J-3})+\cdots+(b_2\times2^0)+\frac{b_1}{2}$$

这样 Q_0 除以 2 的余数是 b_1。将这个计算过程持续下去，可分别得到商序列 $\{Q_k\}$ 和余数序列 $\{b_k\}$。当找到整数 J 满足 $Q_J=0$ 时，计算过程停止。这些序列符合如下公式：

$$\begin{aligned}
N &= 2Q_0 + b_0\\
Q_0 &= 2Q_1 + b_1\\
&\vdots\\
Q_{J-2} &= 2Q_{J-1} + b_{J-1}\\
Q_{J-1} &= 2Q_J + b_J \qquad (Q_J = 0)
\end{aligned} \tag{8}$$

例 1.10　完成 $1563 = 11000011011_{\text{two}}$ 的计算过程。

根据式(8)，从 $N = 1536$ 开始，构造商序列和余数序列：

$$\begin{aligned}
1563 &= 2\times 781+1, & b_0 &= 1\\
781 &= 2\times 390+1, & b_1 &= 1\\
390 &= 2\times 195+0, & b_2 &= 0\\
195 &= 2\times 97+1, & b_3 &= 1\\
97 &= 2\times 48+1, & b_4 &= 1\\
48 &= 2\times 24+0, & b_5 &= 0\\
24 &= 2\times 12+0, & b_6 &= 0\\
12 &= 2\times 6+0, & b_7 &= 0\\
6 &= 2\times 3+0, & b_8 &= 0\\
3 &= 2\times 1+1, & b_9 &= 1\\
1 &= 2\times 0+1, & b_{10} &= 1
\end{aligned}$$

则 1563 的二进制表示为：

$$1563 = b_{10}b_9b_8\cdots b_2b_1b_{0\text{two}} = 11000011011_{\text{two}}$$

1.2.2　序列与级数

当有理数表示为小数形式时，可能需要无穷多的位数。下面是一个常见的例子：

$$\frac{1}{3} = 0.\bar{3} \tag{9}$$

这里的符号 $\bar{3}$ 表示数 3 重复无穷次，形成无穷循环小数。式(9)的基为 10，而且从数学意义上讲，式(9)是式(10)的一个简化表示。

$$\begin{aligned}
S &= (3\times 10^{-1}) + (3\times 10^{-2}) + \cdots + (3\times 10^{-n}) + \cdots\\
&= \sum_{k=1}^{\infty} 3(10)^{-k} = \frac{1}{3}
\end{aligned} \tag{10}$$

如果只显示有限位数，则可得到 1/3 的近似值。例如，$1/3 \approx 0.333 = 333/1000$。近似值的误差为 1/3000。根据式(10)，可验证 $1/3 = 0.333 + 1/3000$。

理解式(10)的展开表示是非常重要的。一个基本的方法是在两边都乘以 10，再相减，如下所示。

$$\begin{aligned}10S &= 3 + (3\times10^{-1}) + (3\times10^{-2}) + \cdots + (3\times10^{-n}) + \cdots \\ -S &= \quad -(3\times10^{-1}) - (3\times10^{-2}) - \cdots - (3\times10^{-n}) - \cdots \\ \hline 9S &= 3 + (0\times10^{-1}) + (0\times10^{-2}) + \cdots + (0\times10^{-n}) + \cdots\end{aligned}$$

这样可得到 $S = 3/9 = 1/3$。在许多有关微积分的书中都可找到证明两个无穷级数之差的相关定理。下面将介绍一些基本概念,要得到更详细的内容可参见与微积分相关的正式教材。

定义 1.6(几何级数) 无穷级数:

$$\sum_{n=0}^{\infty} cr^n = c + cr + cr^2 + \cdots + cr^n + \cdots \tag{11}$$

其中 $c \neq 0$,且 $r \neq 0$,称之为比率为 r 的几何级数。

定理 1.14(几何级数) 几何级数有如下性质:

$$\text{如果} |r| < 1, \text{则} \sum_{n=0}^{\infty} cr^n = \frac{c}{1-r} \tag{12}$$

$$\text{如果} |r| > 1, \text{则级数发散} \tag{13}$$

证明:有限项几何级数的和可表示为:

$$S_n = c + cr + cr^2 + \cdots + cr^n = \frac{c(1-r^{n+1})}{1-r} \quad \text{其中} \quad r \neq 1 \tag{14}$$

为了建立式(12),可以看到:

$$|r| < 1 \text{ 意味着 } \lim_{n\to\infty} r^{n+1} = 0 \tag{15}$$

这样当 $n \to \infty$,根据式(14)和(15)可得:

$$\lim_{n\to\infty} S_n = \frac{c}{1-r}\left(1 - \lim_{n\to\infty} r^{n+1}\right) = \frac{c}{1-r}$$

根据 1.1 节的式(15),可证明(12)成立。

如果 $|r| \geqslant 1$,则序列 $\{r^{n+1}\}$ 不收敛。因此式(14)中的序列 $\{S_n\}$ 不存在极限。这样,可证明式(13)成立。

根据定理 1.14 中的式(12),可得到一个将无穷循环小数转换成分数的有效方法。

例 1.11

$$\begin{aligned}0.\bar{3} &= \sum_{k=1}^{\infty} 3(10)^{-k} = -3 + \sum_{k=0}^{\infty} 3(10)^{-k} \\ &= -3 + \frac{3}{1 - \frac{1}{10}} = -3 + \frac{10}{3} = \frac{1}{3}\end{aligned}$$

1.2.3 二进制分数

二进制(基数为 2)分数可表示为 2 的负幂的和。如果 R 是一个实数,且 $0 < R < 1$,则存在 $d_1, d_2, \cdots, d_n, \cdots$,满足:

$$R = (d_1 \times 2^{-1}) + (d_2 \times 2^{-2}) + \cdots + (d_n \times 2^{-n}) + \cdots \tag{16}$$

其中 $d_j \in \{0,1\}$。式(16)右边的值通常可表示成二进制分数形式:

$$R = 0.d_1 d_2 \cdots d_n \cdots_{\text{two}} \tag{17}$$

许多实数的二进制表示都有无穷位。分数 7/10 的十进制表示为 0.7,但它的二进制表示都有

无穷位：

$$\frac{7}{10}=0.1\overline{0110}_{\text{two}} \tag{18}$$

式(18)中的二进制小数表示有无穷位，由 0110 四个数重复循环构成。

现在可开发一个计算二进制表示的有效算法。如果式(16)的两边都乘以 2，则结果为：

$$2R=d_1+((d_2\times 2^{-1})+\cdots+(d_n\times 2^{-n+1})+\cdots) \tag{19}$$

式(19)右边的括号中的值是一个小于 1 的正数。因此 d_1 是 $2R$ 的整数部分。继续上述步骤，可得到式(19)的小数部分，表示为：

$$F_1=\text{frac}(2R)=(d_2\times 2^{-1})+\cdots+(d_n\times 2^{-n+1})+\cdots \tag{20}$$

其中 $\text{frac}(2R)$是实数 $2R$ 的小数部分。在式(20)两边乘以 2，可得：

$$2F_1=d_2+((d_3\times 2^{-1})+\cdots+(d_n\times 2^{-n+2})+\cdots) \tag{21}$$

根据式(21)的整数部分可得 $d_2=\text{int}(2F_1)$。

如果继续上述过程，无限执行下去（如果 R 是一个无限不重复的二进制表示），则递归地得到序列$\{d_k\}$和$\{F_k\}$：

$$d_k=\text{int}(2F_{k-1}),$$
$$F_k=\text{frac}(2F_{k-1}) \tag{22}$$

其中 $d_1=\text{int}(2R)$，且 $F_1=\text{frac}(2R)$。通过下列收敛的几何级数：

$$R=\sum_{j=1}^{\infty}d_j(2)^{-j}$$

可描述出 R 的二进制小数表示。

例 1.12 通过式(22)，可以计算出式(18)中 7/10 的二进制小数表示。设 $R=7/10=0.7$，则：

$$\begin{array}{lll}
2R=1.4 & d_1=\text{int}(1.4)=1 & F_1=\text{frac}(1.4)=0.4\\
2F_1=0.8 & d_2=\text{int}(0.8)=0 & F_2=\text{frac}(0.8)=0.8\\
2F_2=1.6 & d_3=\text{int}(1.6)=1 & F_3=\text{frac}(1.6)=0.6\\
2F_3=1.2 & d_4=\text{int}(1.2)=1 & F_4=\text{frac}(1.2)=0.2\\
2F_4=0.4 & d_5=\text{int}(0.4)=0 & F_5=\text{frac}(0.4)=0.4\\
2F_5=0.8 & d_6=\text{int}(0.8)=0 & F_6=\text{frac}(0.8)=0.8\\
2F_6=1.6 & d_7=\text{int}(1.6)=1 & F_7=\text{frac}(1.6)=0.6
\end{array}$$

注意 $2F_2=1.6=2F_6$。当 $k=2,3,4,\cdots$时，可得到模式 $d_k=d_{k+4}$和 $F_k=F_{k+4}$。这样 $7/10=0.1\overline{0110}_{\text{two}}$。

通过几何级数可找到二进制有理数的十进制形式。

例 1.13 求解二进制有理数 $0.\overline{01}_{\text{two}}$的十进制形式。根据展开式有：

$$\begin{aligned}0.\overline{01}_{\text{two}}&=(0\times 2^{-1})+(1\times 2^{-2})+(0\times 2^{-3})+(1\times 2^{-4})+\cdots\\
&=\sum_{k=1}^{\infty}(2^{-2})^k=-1+\sum_{k=0}^{\infty}(2^{-2})^k\end{aligned}$$

$$= -1 + \frac{1}{1-\frac{1}{4}} = -1 + \frac{4}{3} = \frac{1}{3}$$

1.2.4 二进制移位

通过移位操作,可以简化求解无穷循环的二进制有理数的过程。例如,S 为:

$$S = 0.00000\overline{11000}_{two} \tag{23}$$

在式(23)两边乘以 2^5,将小数点右移 5 位,得到 $32S$,即:

$$32S = 0.\overline{11000}_{two} \tag{24}$$

同样,在式(23)两边乘以 2^{10},将小数点右移 10 位,得到 $1024S$,即:

$$1024S = 11000.\overline{11000}_{two} \tag{25}$$

式(25)的左右两边减去式(24)的左右两边可得到 $992S = 11000_{two}$ 或根据 $11000_{two} = 24$,得到 $992S = 24$,则 $S = 8/33$。

1.2.5 科学计数法

通过对十进制小数点移位并乘以 10 的幂,是表示实数的标准方法之一,通常被称之为科学计数法。例如:

$$0.0000747 = 7.47 \times 10^{-5}$$
$$31.4159265 = 3.14159265 \times 10$$
$$9\ 700\ 000\ 000 = 9.7 \times 10^9$$

在化学领域,阿伏伽德罗(Avogadro)数是一个重要的常量,可表示为 6.02252×10^{23}。它是单个分子的克原子重量下的原子个数。在计算机中,$1K = 1.024 \times 10^3$。

1.2.6 机器数

计算机使用规格化浮点二进制表示实数。这意味着实数 x 的算术值并没有实际存放在计算机中,计算机实际存放的是 x 的近似值:

$$x \approx \pm q \times 2^n \tag{26}$$

其中数 q 称为尾数,它是一个有限的二进制数,满足不等式 $1/2 \leqslant q < 1$。整数 n 称为阶码。

在计算机中,只使用了实数系统的一小部分子集。典型的子集一般只包含式(26)表示的一部分。二进制数的个数受 q 和 n 的严格限制。例如,在下述正实数集合中的所有实数:

$$0.d_1d_2d_3d_{4two} \times 2^n \tag{27}$$

其中 $d_1 = 1$,而 d_2, d_3, d_4 为 0 或 1,而且 $n \in \{-3,-2,-1,0,1,2,3,4\}$。在式(27)中尾数和阶码各有 8 种选择,可产生 64 个数的集合:

$$\{0.1000_{two} \times 2^{-3}, 0.1001_{two} \times 2^{-3}, \cdots, 0.1110_{two} \times 2^4, 0.1111_{two} \times 2^4\} \tag{28}$$

这 64 个数的十进制形式如表 1.3 所示。当式(27)中的尾数和阶码受限时,计算机只能选用有限的数存储实数 x 的近似值。

表1.3 尾数为4比特(bit),阶码为 $n=-3,-2,\cdots,3,4$ 的二进制数集合的十进制表示

尾数	阶码							
	$n=-3$	$n=-2$	$n=-1$	$n=0$	$n=1$	$n=2$	$n=3$	$n=4$
0.1000_{two}	0.0625	0.125	0.25	0.5	1	2	4	8
0.1001_{two}	0.0703125	0.140625	0.28125	0.5625	1.125	2.25	4.5	9
0.1010_{two}	0.078125	0.15625	0.3125	0.625	1.25	2.5	5	10
0.1011_{two}	0.0859375	0.171875	0.34375	0.6875	1.375	2.75	5.5	11
0.1100_{two}	0.09375	0.1875	0.375	0.75	1.5	3	6	12
0.1101_{two}	0.1015625	0.203125	0.40625	0.8125	1.625	3.25	6.5	13
0.1110_{two}	0.109375	0.21875	0.4375	0.875	1.75	3.5	7	14
0.1111_{two}	0.1171875	0.234375	0.46875	0.9375	1.875	3.75	7.5	15

如果计算机的尾数为4比特,如何计算$\left(\frac{1}{10}+\frac{1}{5}\right)+\frac{1}{6}$?假定计算机将实数四舍五入为表1.3中的二进制数,可根据表1.3查找各个实数所对应的最佳近似值:

$$\frac{1}{10}\approx 0.1101_{two}\times 2^{-3}=0.01101_{two}\times 2^{-2}$$

$$\frac{\frac{1}{5}\approx 0.1101_{two}\times 2^{-2}=\ \ 0.1101_{two}\times 2^{-2}}{\frac{3}{10}\qquad\qquad\qquad\qquad 1.00111_{two}\times 2^{-2}}\tag{29}$$

计算机必须确定如何存储数 $1.00111_{two}\times 2^{-2}$。若数被四舍五入为 $0.1010_{two}\times 2^{-1}$,则下一步是:

$$\frac{3}{10}\approx 0.1010_{two}\times 2^{-1}=0.1010_{two}\times 2^{-1}$$

$$\frac{\frac{1}{6}\approx 0.1011_{two}\times 2^{-2}=0.01011_{two}\times 2^{-1}}{\frac{7}{15}\qquad\qquad\qquad\qquad 0.11111_{two}\times 2^{-1}}\tag{30}$$

计算机必须确定如何存储数 $0.11111_{two}\times 2^{-1}$。由于假定数被四舍五入,则数被存储为 $0.10000_{two}\times 2^{0}$。因此,加法问题的计算机解为:

$$\frac{7}{15}\approx 0.10000_{two}\times 2^{0}\tag{31}$$

计算机的计算误差为:

$$\frac{7}{15}-0.10000_{two}\approx 0.466667-0.500000\approx 0.033333\tag{32}$$

如果用百分比来表示,则误差为7.14%。

1.2.7 计算机精度

为了精确地存储数值,计算机必须使用二进制浮点数。其中,要表示7位十进制数至少需要24比特的尾数;如果使用32比特的尾数,则可存储9位十进制数。现在重新考虑在本节开始时式(1)中遇到的困难,即计算机累加1/10。

设式(26)中的尾数 q 的位数为32比特。根据条件 $1/2\leqslant q$,可得到第一个数 $d_1=1$。因

此,q 有如下形式:

$$q = 0.1d_2d_3\cdots d_{31}d_{32\text{two}} \tag{33}$$

当把小数部分用二进制形式表示时,通常是无限循环小数。例如下列表达式:

$$\frac{1}{10} = 0.0\overline{0011}_{\text{two}} \tag{34}$$

当使用 32 比特的尾数时,计算机对数进行截断后的内在近似值为:

$$\frac{1}{10} \approx 0.11001100110011001100110011001100_{\text{two}} \times 2^{-3} \tag{35}$$

式(35)中近似值的误差为式(35)与式(34)的差值,即:

$$0.\overline{1100}_{\text{two}} \times 2^{-35} \approx 2.328306437 \times 10^{-11} \tag{36}$$

由于式(36)的存在,计算机对式(1)中的 1/10 进行 100 000 次累加后,必然存在误差。误差大于$(100\ 000)(2.328306437\times10^{-11}) = 2.328306437\times10^{-6}$。实际上,在计算过程中还存在着更大的误差。原因是部分和可能随时会被四舍五入,而且随着和的增加,1/10 的后一个加数远小于逐渐增加的累加和,它在计算中的影响会被舍弃。这样各种误差带来的组合效应使得实际产生的误差为 $10\ 000 - 9999.99447 = 5.53\times10^{-3}$。

1.2.8 计算机浮点数

计算机采用整数模式和浮点模式表示数。整数模式主要用来进行整数计算,在数值分析中的作用有限。而科学工程应用中主要采用浮点数。必须要明确如下事实:在计算机实现式(26)时,尾数 q 的位数是有限的,而且阶码 n 的范围肯定也是有限的。

在以 32 比特表示单精度实数的计算机中,阶码用 8 比特表示,尾数用 24 比特表示。因此,它可表示的实数范围是:

$$2.938736E-39 \quad 到 \quad 1.701412E+38$$

(即 2^{-128}到 2^{127}),有 6 位十进制数值精度(例如,$2^{-23} = 1.2\times10^{-7}$)。

在以 48 比特表示单精度实数的计算机中,阶码用 8 比特表示,尾数用 40 比特表示。因此,它可表示的实数范围是:

$$2.9387358771E-39 \quad 到 \quad 1.7014118346E+38$$

(即 2^{-128}到 2^{127}),有 11 位十进制数值精度(例如,$2^{-39} = 1.8\times10^{-12}$)。

对于以 64 比特表示双精度实数的计算机,它可能用 11 比特表示阶码,用 53 比特表示尾数。因此,它可表示的实数范围是:

$$5.562684646268003E-309 \quad 到 \quad 8.988465674311580E+307$$

(即 2^{-1024}到 2^{1023}),有 16 位十进制数值精度(例如,$2^{-52} = 2.2\times10^{-16}$)。

1.2.9 二进制数的练习

1. 利用计算机求解下列表达式。提示:目的是让计算机重复执行减法计算,不要借助于乘法计算。

   (a) $10\ 000 - \sum_{k=1}^{100\ 000} 0.1$ (b) $10\ 000 - \sum_{k=1}^{80\ 000} 0.125$

2. 利用式(4)和式(5),将下列二进制数转换成十进制形式。

   (a) 10101_{two} (b) 111000_{two}

(c) 11111110_{two}　　(d) 1000000111_{two}

3. 利用式(16)和式(17),将下列二进制小数转换成十进制形式。

(a) 0.11011_{two}　　(b) 0.10101_{two}

(c) 0.1010101_{two}　　(d) 0.110110110_{two}

4. 将下列二进制数转换成十进制形式。

(a) 1.0110101_{two}　　(b) 11.0010010001_{two}

5. 练习 4 中的数是$\sqrt{2}$和 π 的近似值。求解近似值的误差,即计算下列值。

(a) $\sqrt{2}-1.0110101_{two}$(利用$\sqrt{2}=1.41421356237309\cdots$)

(b) $\pi-11.0010010001_{two}$(利用 $\pi=3.14159265358979\cdots$)

6. 参照例 1.10,将下列十进制数转换成二进制形式。

(a) 23　　(b) 87　　(c) 378　　(d) 238 8

7. 参照例 1.12,将下列十进制数转换成如 $0.d_1d_2\cdots d_{n\,two}$形式的二进制小数。

(a) 7/16　　(b) 13/16　　(c) 23/32　　(d) 75/128

8. 参照例 1.12,将下列十进制数转换成二进制无限循环小数。

(a) 1/10　　(b) 1/3　　(c) 1/7

9. 求解下列具有 7 位有效数字的二进制近似值的误差 $R-0.d_1d_2d_3d_4d_5d_6d_{7\,two}$。

(a) $1/10\approx0.0001100_{two}$　　(b) $1/7\approx0.0010010_{two}$

10. 证明二进制展开式 $1/7=0.\overline{001}_{two}$与$\frac{1}{7}=\frac{1}{8}+\frac{1}{64}+\frac{1}{512}+\cdots$等价。利用定理 1.14 来建立这个展开式。

11. 证明二进制展开式 $1/5=\overline{0011}_{two}$与$\frac{1}{5}=\frac{3}{16}+\frac{3}{256}+\frac{1}{4096}+\cdots$等价。利用定理 1.14 来建立这个展开式。

12. 证明对于任意的数 2^{-N},其中 N 是正整数,可表示为 N 位十进制数,即 $2^{-N}=0.d_1d_2d_3\cdots d_N$。提示:1/2=0.5,1/4=0.25,…。

13. 根据表 1.3 来判断当用有 4 位尾数的计算机执行下列计算时,会得到什么结果。

(a) $\left(\frac{1}{3}+\frac{1}{5}\right)+\frac{1}{6}$　　(b) $\left(\frac{1}{10}+\frac{1}{3}\right)+\frac{1}{5}$

(c) $\left(\frac{3}{17}+\frac{1}{9}\right)+\frac{1}{7}$　　(d) $\left(\frac{7}{10}+\frac{1}{9}\right)+\frac{1}{7}$

14. 证明当将式(8)中所有等式中的 2 替换为 3 时,可得到一个求解正整数的三进制表示形式的方法,并求解下列整数的三进制表示形式。

(a) 10　　(b) 23　　(c) 421　　(d) 178 4

15. 证明当将式(22)中所有等式中的 2 替换为 3 时,可得到一个求解正数 $R(0<R<1)$的三进制表示形式的方法,并求解下列正数的三进制表示形式。

(a) 1/3　　(b) 1/2　　(c) 1/10　　(d) 11/27

16. 证明当将式(8)中的所有等式中的 2 替换为 5 时,可得到一个求解正整数的五进制表示形式的方法,并求解下列整数的五进制表示形式。

(a) 10　　(b) 35　　(c) 721　　(d) 734

17. 证明当将式(22)中的所有等式中的 2 替换为 5 时,可得到一个求解正数 $R(0<R<1)$ 的五进制表示形式的方法,并求解下列正数的五进制表示形式。
(a) 1/3 (b) 1/2 (c) 1/10 (d) 154/625

1.3 误差分析

读者将从数值分析的练习中学习到非常重要的一点,即计算结果并不确切地等于精确结果,因为存在不少隐含的方法会破坏数值结果的精度。对这一点的理解将有助于研究人员实现和开发正确的数值算法。

定义 1.7 设 $\hat{p}$ 是 p 的近似值,则绝对误差是 E_p,简称误差。相对误差是 $R_p=|p-\hat{p}|/|p|$,其中 $p\neq 0$。

误差仅仅是真值与近似值之间的差,而相对误差在很大程度上依赖于真值。

例 1.14 找出下面三种情况下的误差和相对误差。

设 $x=3.141592, \hat{x}=3.14$,则误差为:

$$E_x=|x-\hat{x}|=|3.141592-3.14|=0.001592 \tag{1a}$$

相对误差为:

$$R_x=\frac{|x-\hat{x}|}{|x|}=\frac{0.001592}{3.141592}=0.00507$$

设 $y=1\,000\,000, \hat{y}=999\,996$,则误差为:

$$E_y=|y-\hat{y}|=|1\,000\,000-999\,996|=4 \tag{1b}$$

相对误差为:

$$R_y=\frac{|y-\hat{y}|}{|y|}=\frac{4}{1\,000\,000}=0.000004$$

设 $z=0.000012, \hat{z}=0.000009$,则误差为:

$$E_z=|z-\hat{z}|=|0.000012-0.000009|=0.000003 \tag{1c}$$

相对误差为:

$$R_z=\frac{|z-\hat{z}|}{|z|}=\frac{0.000003}{0.000012}=0.25$$

在例(1a)中,E_x 和 R_x 间无太大差别,都可用来判别 $\hat{x}$ 的精度。在例(1b)中,y 值的数量级为 10^6,误差 E_y 很大,相对误差很小。在这种情况下,可认为 $\hat{y}$ 是 y 较好的近似值。在例(1c)中,z 值的数量级为 10^{-6},误差 E_z 是 3 种情况中最小的,但相对误差是最大的。如果用百分数的形式,相对误差为 25%,这样 $\hat{z}$ 是 z 的不良近似值。当 $|p|$ 远离 1 时(大于或小于),与误差 E_p 相比,相对误差 R_p 能更好地表示近似值的精确程度。由于相对误差直接处理尾数,所以浮点形示主要采用相对误差。

定义 1.8 如果 d 是满足下列不等式的最大正整数,则称数 $\hat{p}$ 近似 p 时具有 d 位有效数字。

$$\frac{|p-\hat{p}|}{|p|}<\frac{10^{-d}}{2} \tag{2}$$

例 1.15 判断例 1.14 中近似值的有效数字。

如果 $x = 3.141592, \hat{x} = 3.14$，则 $|x - \hat{x}|/|x| = 0.000507 < 10^{-2}/2$。因此，$\hat{x}$ 近似 x 时具有2位有效数字。 (3a)

如果 $y = 1\ 000\ 000, \hat{y} = 999\ 996$，则 $|y - \hat{y}|/|y| = 0.000004 < 10^{-5}/2$。因此，$\hat{y}$ 近似 y 时具有5位有效数字。 (3b)

如果 $z = 0.000012, \hat{z} = 0.000009$，则 $|z - \hat{z}|/|z| = 0.25 < 10^{-0}/2$。因此，$\hat{z}$ 近似 z 时无有效数字。 (3c)

1.3.1 截断误差

截断误差通常是指用一个基本表达式替换一个相当复杂的算术表达式时所引入的误差。这一术语从用截断泰勒级数替换一个复杂表达式的技术中衍生而来。例如，无穷泰勒级数：

$$e^{x^2} = 1 + x^2 + \frac{x^4}{2!} + \frac{x^6}{3!} + \frac{x^8}{4!} + \cdots + \frac{x^{2n}}{n!} + \cdots$$

可用它的前5项 $1 + x^2 + \frac{x^4}{2!} + \frac{x^6}{3!} + \frac{x^8}{4!}$ 来替代。这样可以完成求解累积的数值近似值。

例 1.16 设有 $\int_0^{1/2} e^{x^2}\,dx = 0.544987104184 = p$，当用截断泰勒级数 $P_8(x) = 1 + x^2 + \frac{x^4}{2!} + \frac{x^6}{3!} + \frac{x^8}{4!}$ 替代被积函数 $f(x) = e^{x^2}$ 时，确定积分近似值的精度。

替换后的积分近似值为：

$$\int_0^{1/2}\left(1 + x^2 + \frac{x^4}{2!} + \frac{x^6}{3!} + \frac{x^8}{4!}\right)dx = \left(x + \frac{x^3}{3} + \frac{x^5}{5(2!)} + \frac{x^7}{7(3!)} + \frac{x^9}{9(4!)}\right)\Bigg|_{x=0}^{x=1/2}$$

$$= \frac{1}{2} + \frac{1}{24} + \frac{1}{320} + \frac{1}{5376} + \frac{1}{110\ 592}$$

$$= \frac{2\ 109\ 491}{3\ 870\ 720} = 0.544986720817 = \hat{p}$$

因为 $10^{-5}/2 > |p - \hat{p}|/|p| = 7.03442 \times 10^{-7} > 10^{-6}/2$，则近似值 $\hat{p}$ 近似真值 $p = 0.544987104184$ 时有5位有效数字。$y = f(x) = e^{x^2}$，$y = P_8(x)$ 的曲线图，范围在 $0 \leq x \leq 1/2$ 内的曲线下区域面积如图 1.7 所示。

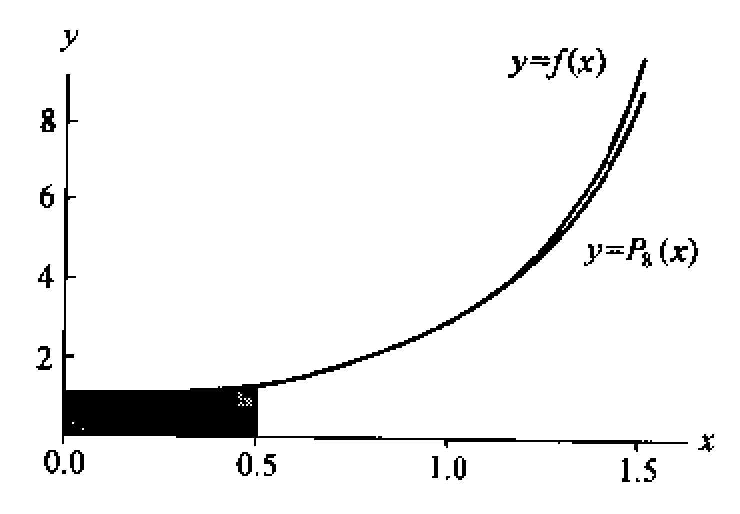


图 1.7 $y = f(x) = e^{x^2}$，$y = P_8(x)$ 的函数曲线图，以及当 $0 \leq x \leq \frac{1}{2}$ 时曲线下的区域面积

1.3.2 舍入误差

计算机表示的实数受限于尾数的固定精度，因此有时并不能确切地表示真值，这一类型的

误差被称为舍入误差。在上一节中，计算机对存储的实数 $1/10 = 0.0\overline{0011}_{two}$ 进行了截断，对实际存储在计算机中的数的最后一位进行了四舍五入。综上所述，由于计算机硬件只支持有限位机器数的运算，因此在计算中将可能引入和传播舍入误差。

1.3.3 舍去和舍入

设任意实数 p 的规格化浮点表示形式为：

$$p = \pm 0.d_1 d_2 d_3 \cdots d_k d_{k+1} \cdots \times 10^n \tag{4}$$

其中当 $j > 1$ 时，有 $1 \leqslant d_1 \leqslant 9$ 和 $0 \leqslant d_j \leqslant 9$。设 k 是计算机浮点计算中浮点数的最大位数，则实数 p 表示为 $fl_{chop}(p)$：

$$fl_{chop}(p) = \pm 0.d_1 d_2 d_3 \cdots d_k \times 10^n \tag{5}$$

其中当 $1 < j \leqslant k$ 时，有 $0 \leqslant d_j \leqslant 9$ 和 $1 \leqslant d_1 \leqslant 9$。数 $fl_{chop}(p)$ 称为 p 的舍去浮点表示(chopped floating - pointing representation)。在这种情况下，$fl_{chop}(p)$ 的第 k 位数与 p 的第 k 位数相同。另一种 k 位表示法是舍入浮点表示(rounded floating - point representation) $fl_{round}(p)$：

$$fl_{round}(p) = \pm 0.d_1 d_2 d_3 \cdots r_k \times 10^n \tag{6}$$

其中当 $1 < j < k$ 时，有 $1 \leqslant d_1 \leqslant 9$ 和 $0 \leqslant d_j \leqslant 9$，而且最后一位数 r_k 是最逼近 $d_k d_{k+1} d_{k+2} \cdots$ 的整数。例如，有实数：

$$p = \frac{22}{7} = 3.142857142857142857\cdots$$

则有如下的 6 位有效数字表示形式：

$$fl_{chop}(p) = 0.314285 \times 10^1$$

$$fl_{round}(p) = 0.314286 \times 10^1$$

通常，p 的舍去表示和舍入表示可分别写成 3.14285 和 3.14286。读者必须从根本上了解和注意计算机使用的各种舍入浮点表示法。

1.3.4 精度损失

设有两个数 $p = 3.1415926536$ 和 $p = 3.1415957341$，二者几乎相等，且都有 11 位有效数字精度。它们的差为：$p - q = -0.0000030805$。由于 p 和 q 的前 6 位数相等，所以它们的差别只有 5 位数字精度。这种现象称为精度损失(loss of significance)或差额抵消(subtractive cancellation)。如果不注意到这一点，最终计算结果的精度可能会逐渐减少。

例 1.17 设 $f(x) = x(\sqrt{x+1} - \sqrt{x})$，$g(x) = \dfrac{x}{\sqrt{x+1} + \sqrt{x}}$。用 6 位有效数字和舍入法比较 $f(500)$ 和 $g(500)$ 的计算结果。

先考虑第一个函数：

$$\begin{aligned} f(500) &= 500(\sqrt{501} - \sqrt{500}) \\ &= 500(22.3830 - 22.3607) = 500(0.0223) = 11.1500 \end{aligned}$$

对于 $g(x)$ 有：

$$g(500) = \frac{500}{\sqrt{501} + \sqrt{500}}$$

$$= \frac{500}{22.3830 + 22.3607} = \frac{500}{44.7437} = 11.1748$$

从数学意义上讲，第二个函数 $g(x)$ 等价于 $f(x)$，其推导过程如下所示：

$$f(x) = \frac{x(\sqrt{x+1} - \sqrt{x})(\sqrt{x+1} + \sqrt{x})}{\sqrt{x+1} + \sqrt{x}}$$

$$= \frac{x((\sqrt{x+1})^2 - (\sqrt{x})^2)}{\sqrt{x+1} + \sqrt{x}}$$

$$= \frac{x}{\sqrt{x+1} + \sqrt{x}}$$

结果 $g(500) = 11.1748$ 存在较少的误差，当真值 11.174755399747198…舍入到 6 位数时与 $g(500)$ 的值相同。

读者可通过习题 12 学习如何避免在二次根公式中的精度损失。下例显示的截断泰勒级数有时有助于避免精度损失造成的误差。

例 1.18 设

$$f(x) = \frac{e^x - 1 - x}{x^2}, P(x) = \frac{1}{2} + \frac{x}{6} + \frac{x^2}{24}$$

用舍入法和 6 位有效数字比较 $f(0.01)$ 和 $P(0.01)$ 的计算结果。函数 $P(x)$ 是 $f(x)$ 在 $x = 0$ 处的展开的二阶泰勒多项式。

对于第一个函数有：

$$f(0.01) = \frac{e^{0.01} - 1 - 0.01}{(0.01)^2} = \frac{1.010050 - 1 - 0.01}{0.001} = 0.5$$

对于第二个函数有：

$$P(0.01) = \frac{1}{2} + \frac{0.01}{6} + \frac{0.001}{24}$$

$$= 0.5 + 0.001667 + 0.000004 = 0.501671$$

结果 $P(0.01) = 0.501671$ 存在较少的误差，当真值 0.50167084168057542…舍入到 6 位数时与 $g(500)$ 的值相同。

对于多项式求值计算，有时可以通过将表达式重新表示为嵌套乘的形式，以获得较为理想的结果。

例 1.19 设 $P(x) = x^3 - 3x^2 + 3x - 1$ 和 $Q(x) = ((x - 3)x + 3)x - 1$。用 3 位舍入算法计算 $P(2.19)$ 和 $Q(2.19)$，并与真值进行比较，其中 $P(2.19) = Q(2.19) = 1.685159$。

$$P(2.19) \approx (2.19)^3 - 3(2.19)^2 + 3(2.19) - 1$$

$$= 10.5 - 14.4 + 6.57 - 1 = 1.67$$

$$Q(2.19) \approx ((2.19 - 3)2.19 + 3)2.19 - 1 = 1.69$$

误差分别为 0.015159 和 −0.004841，则近似值 $Q(2.19) \approx 1.69$ 的误差较小。习题 6 探讨了当接近多项式根时的求解情况。

1.3.5 $O(h^n)$阶逼近

序列$\left\{\frac{1}{n^2}\right\}_{n=1}^{\infty}$和$\left\{\frac{1}{n}\right\}_{n=1}^{\infty}$都明显地收敛到0。而且第一个序列比第二个序列收敛得快。在接下来的章节中,会使用一些特殊的术语和表示以描述序列如何快速收敛。

定义 1.9 设有函数$f(h)$和$g(h)$,如果存在常数C和c,使得:

对任意$h\leqslant c$,有 $$|f(h)|\leqslant C|g(h)| \tag{7}$$

则称函数$f(h)$为 ***big Oh*** 函数$g(h)$,表示为$f(h)=O(g(h))$。

例 1.20 设有函数$f(x)=x^2+1$和$g(x)=x^3$,由于当$x^2\leqslant x^3$时,有$1\leqslant x^3$且$x\geqslant 1$,因此可推导出当$x\geqslant 1$时,有$x^2+1\leqslant 2x^3$。因此$f(x)=O(g(x))$。

对于常见的基本函数(x^n,$x^{1/n}$,a^x,$\log_a{}^x$等),符号"big Oh"表示提供了一个描述函数增长率的有效方法。

用类似的方法也可描述序列的收敛率。

定义 1.10 设有两个序列$\{x_n\}_{n=1}^{\infty}$和$\{y_n\}_{n=1}^{\infty}$。如果存在常数C和N,使得:

对任意$n\geqslant N$,有 $$|x_n|\leqslant C|y_n| \tag{8}$$

则称序列$\{x_n\}$以序列$\{y_n\}$为上界,并记为:$x_n=O(y_n)$。

例 1.21 因为对任意$n\geqslant 1$,有$\frac{n^2-1}{n^3}\leqslant\frac{n^2}{n^3}=\frac{1}{n}$,所以$\frac{n^2-1}{n^3}=O\left(\frac{1}{n}\right)$。

通常用函数$p(h)$近似函数$f(h)$,且误差界表示为$M|h^n|$,这引出下述定义。

定义 1.11 设函数$f(h)$的近似为$p(h)$,且存在实常数$M>0$和正整数n,满足:

$$\frac{|f(h)-p(h)|}{|h^n|}\leqslant M,\text{ 当 } h \text{ 足够小时} \tag{9}$$

则称$p(h)$以近似阶$O(h^n)$来近似$f(h)$,表示为:

$$f(h)=p(h)+O(h^n) \tag{10}$$

将式(9)重写为$|f(h)-p(h)|\leqslant M|h^n|$,可以看到,符号$O(h_n)$代表误差界$M|h^n|$。接下来的结论可用来定义两个函数的四则运算。

定理 1.15 设$f(h)=p(h)+O(h_n)$,$g(h)=q(h)+O(h_m)$,且$r=\min\{m,n\}$,则:

$$f(h)+g(h)=p(h)+q(h)+O(h^r) \tag{11}$$

$$f(h)g(h)=p(h)q(h)+O(h^r) \tag{12}$$

而且:

$$\frac{f(h)}{g(h)}=\frac{p(h)}{q(h)}+O(h^r),\text{其中 } g(h)\neq 0,\text{且 } q(h)\neq 0 \tag{13}$$

当将$p(x)$看做是$f(x)$的第n个泰勒多项式近似值时,余项可指定为$O(h^{n+1})$,这代表被忽略的从幂h^{n+1}开始的项。余项收敛到0的速度与h^{n+1}收敛到0的速度一样,它们的关系可表示为:

$$O(h^{n+1}) \approx Mh^{n+1} \approx \frac{f^{(n+1)}(c)}{(n+1)!}h^{n+1} \tag{14}$$

这里 h 要足够小。因此符号 $O(h^{n+1})$可用 Mh^{n+1}表示,其中 M 是一个常数,或“可理解为常数”。

定理 1.16(泰勒定理) 设$f \in C^{n+1}[a,b]$内,如果 x_0 和 $x = x_0 + h$ 都在区间$[a,b]$内,则:

$$f(x_0 + h) = \sum_{k=0}^{n} \frac{f^{(k)}(x_0)}{k!}h^k + O(h^{n+1}) \tag{15}$$

下面的例子阐明了上述定理。计算过程使用了加法性质(i)$O(h^p) + O(h^p) = O(h^p)$,(ii)$O(h^p) + O(h^q) = O(h^r)$,其中 $r = \min\{p,q\}$,和乘法性质(iii)$O(h^p)O(h^q) = O(h^s)$,其中 $s = p + q$。

例 1.22 考虑如下泰勒多项式的展开:

$$e^h = 1 + h + \frac{h^2}{2!} + \frac{h^3}{3!} + O(h^4) \text{和} \cos(h) = 1 - \frac{h^2}{2!} + \frac{h^4}{4!} + O(h^6)$$

确定它们的和与积的近似阶。

对于求和有:

$$e^h + \cos(h) = 1 + h + \frac{h^2}{2!} + \frac{h^3}{3!} + O(h^4) + 1 - \frac{h^2}{2!} + \frac{h^4}{4!} + O(h^6)$$

$$= 2 + h + \frac{h^3}{3!} + O(h^4) + \frac{h^4}{4!} + O(h^6)$$

因为 $O(h^4) + \frac{h^4}{4!} = O(h^4)$和 $O(h^4) + O(h^6) = O(h^4)$,上式可简化为:

$$e^h + \cos(h) = 2 + h + \frac{h^3}{3!} + O(h^4)$$

近似阶为 $O(h^4)$。

对于乘积的处理和求和类似,如下所示:

$$e^h\cos(h) = \left(1 + h + \frac{h^2}{2!} + \frac{h^3}{3!} + O(h^4)\right)\left(1 - \frac{h^2}{2!} + \frac{h^4}{4!} + O(h^6)\right)$$

$$= \left(1 + h + \frac{h^2}{2!} + \frac{h^3}{3!}\right)\left(1 - \frac{h^2}{2!} + \frac{h^4}{4!}\right)$$

$$+ \left(1 + h + \frac{h^2}{2!} + \frac{h^3}{3!}\right)O(h^6) + \left(1 - \frac{h^2}{2!} + \frac{h^4}{4!}\right)O(h^4) + O(h^4)O(h^6)$$

$$= 1 + h - \frac{h^3}{3} - \frac{5h^4}{24} - \frac{h^5}{24} + \frac{h^6}{48} + \frac{h^7}{144} + O(h^6) + O(h^4) + O(h^4)O(h^6)$$

由于 $O(h^4)O(h^6) = O(h^{10})$,并且:

$$-\frac{5h^4}{24} - \frac{h^5}{24} + \frac{h^6}{48} + \frac{h^7}{144} + O(h^6) + O(h^4) + O(h^{10}) = O(h^4)$$

上述表达式可简化为:

$$e^h\cos(h) = 1 + h - \frac{h^3}{3} + O(h^4)$$

这样近似阶为 $O(h^4)$。

1.3.6 序列的收敛阶

通过计算一系列逐渐接近需求解的近似值可进行数值逼近。定义 1.10 中给出了针对序列上界的定义，而序列收敛阶的定义与函数的近似阶的定义 1.11 类似。

定义 1.12 设$\lim_{n\to\infty} x_n = x$，有序列$\{r_n\}_{n=1}^{\infty}$，且$\lim_{n\to\infty} r_n = 0$。如果存在常量 $K>0$，满足：

$$\frac{|x_n - x|}{|r_n|} \leqslant K, n \text{ 足够大}$$

则称$\{x_n\}_{n=1}^{\infty}$以收敛阶 $\boldsymbol{O}(r_n)$收敛于 x。

可表示为 $x_n = x + \boldsymbol{O}(r_n)$，或表示为 $x_n \to x$，收敛阶为 $\boldsymbol{O}(r_n)$。

例 1.23 设有 $x_n = \cos(n)/n^2$ 和 $r_n = 1/n^2$，则$\lim_{n\to\infty} x_n = 0$，收敛阶为 $O(1/n^2)$。根据如下关系式可马上推导出上述结论：

$$\frac{|\cos(n)/n^2|}{|1/n^2|} = |\cos(n)| \leqslant 1 \qquad \text{对所有的 } n$$

1.3.7 误差传播

下面研究在连续计算过程中误差是如何传播的。考虑数 p 和 q(真值)的加法运算，它们的近似值分别为 $\hat{p}$ 和$\hat{q}$，误差分别为 ϵ_p 和ϵ_q。从 $p = \hat{p} + \epsilon_p$ 和$q = \hat{q} + \epsilon_q$ 开始，它们的和为：

$$p + q = (\hat{p} + \epsilon_p) + (\hat{q} + \epsilon_q) = (\hat{p} + \hat{q}) + (\epsilon_p + \epsilon_q) \tag{16}$$

因此对于加法运算，整个和的误差是每个加数的误差的和。

在乘积计算过程中，误差的传播将更为复杂。乘积表达式如下所示：

$$pq = (\hat{p} + \epsilon_p)(\hat{q} + \epsilon_q) = \hat{p}\hat{q} + \hat{p}\epsilon_q + \hat{q}\epsilon_q + \epsilon_p\epsilon_q \tag{17}$$

因此，如果 $\hat{p}$ 和$\hat{q}$ 的绝对值大于 1，则原来的误差 $\hat{p}\epsilon_q$ 和$\hat{q}\epsilon_p$ 会被放大成ϵ_p 和ϵ_q。如果观察相对误差，可以得到更深入的了解。重新排列式(17)中的项可得到：

$$pq - \hat{p}\hat{q} = \hat{p}\epsilon_q + \hat{q}\epsilon_p + \epsilon_p\epsilon_q \tag{18}$$

假定 $p\neq 0$，且 $q\neq 0$，则用 pq 除式(18)可得 pq 乘积的相对误差：

$$R_{pq} = \frac{pq - \hat{p}\hat{q}}{pq} = \frac{\hat{p}\epsilon_q + \hat{q}\epsilon_p + \epsilon_p\epsilon_q}{pq} = \frac{\hat{p}\epsilon_q}{pq} + \frac{\hat{q}\epsilon_p}{pq} + \frac{\epsilon_p\epsilon_q}{pq} \tag{19}$$

进一步假设 $\hat{p}$ 和$\hat{q}$ 是p 和q 较好的近似，则 $\hat{p}/p \approx 1$，$\hat{q}/q \approx 1$，且 $R_pR_q = (\epsilon_p/p)(\epsilon_q/q) \approx 0$($R_p$ 和 R_q 是近似值$\hat{p}$ 和$\hat{q}$ 的相对误差)。将它们替换到式(19)中可得到如下简化的关系式：

$$R_{pq} = \frac{pq - \hat{p}\hat{q}}{pq} \approx \frac{\epsilon_q}{q} + \frac{\epsilon_p}{p} + 0 = R_q + R_p \tag{20}$$

这表明乘积 pq 的相对误差大致等于$\hat{p}$ 和$\hat{q}$ 相对误差的和。

初始误差通常通过一系列的计算进行传播。对任何数值计算而言，都要尽量减少初始误差，因为初始条件下的小误差对最终结果产生的影响较小。这样的算法称为稳定的算法；否则，称为不稳定的算法。在数值分析中，应当尽量选用稳定的算法。下述定义将用来描述误差的传播。

定义 1.13 设 ϵ 表示初始误差，$\epsilon(n)$表示第 n 步计算后的误差增长。如果$|\epsilon(n)| \approx n\epsilon$，则称误差按线性增长。如果$|\epsilon(n)| \approx K^n\epsilon$，则称误差按指数增长。如果 $K>1$，则当 $n\to\infty$时，误差的指数无界增长；如果 $0<k<1$，则当 $n\to\infty$时，误差的指数增长趋于 0。

下两个例子显示了初始误差可稳定传播或不稳定传播。在第一个例子中,介绍了 3 个算法。每个算法递归生成同一序列。在第二个例子中,将对初始条件进行一些小改动,同时对误差传播进行分析。

例 1.24 用无限精度算法结合下列 3 个方案可递归生成序列 $\{1/3^n\}_{n=0}^{\infty}$ 中的各项值。

$$r_0 = 1 \text{ 且 } r_n = \frac{1}{3} r_{n-1}, \qquad \text{当 } n = 1, 2, \cdots \tag{21a}$$

$$p_0 = 1, p_1 = \frac{1}{3} \text{ 且 } p_n = \frac{4}{3} p_{n-1} - \frac{1}{3} p_{n-2}, \qquad \text{当 } n = 2, 3, \cdots \tag{21b}$$

$$q_0 = 1, q_1 = \frac{1}{3} \text{ 且 } q_n = \frac{10}{3} q_{n-1} - q_{n-2}, \qquad \text{当 } n = 2, 3, \cdots \tag{21c}$$

式(21a)的意义很明显。在(21b)中,差分方程的通解是 $p_n = A(1/3^n) + B$。这可以通过直接替换方法进行验证:

$$\frac{4}{3} p_{n-1} - \frac{1}{3} p_{n-2} = \frac{4}{3}\left(\frac{A}{3^{n-1}} + B\right) - \frac{1}{3}\left(\frac{A}{3^{n-2}} + B\right)$$
$$= \left(\frac{4}{3^n} - \frac{3}{3^n}\right)A - \left(\frac{4}{3} - \frac{1}{3}\right)B = A\frac{1}{3^n} + B = p_n$$

设 $A = 1$ 且 $B = 0$,则可得到期望的序列。在(21c)中,差分方程的通解是 $q_n = A(1/3^n) + B3^n$。同样可通过直接替换方法进行验证:

$$\frac{10}{3} q_{n-1} - q_{n-2} = \frac{10}{3}\left(\frac{A}{3^{n-1}} + B3^{n-1}\right) - \left(\frac{A}{3^{n-2}} + B3^{n-2}\right)$$
$$= \left(\frac{10}{3^n} - \frac{9}{3^n}\right)A - (10 - 1)3^{n-2}B$$
$$= A\frac{1}{3^n} + B3^n = q_n$$

设 $A = 1$ 且 $B = 0$,则可得到期望的序列。

例 1.25 用下列方案求出序列 $\{x_n\} = \{1/3^n\}$ 的近似值:

$$r_0 = 0.99996 \text{ 且 } r_n = \frac{1}{3} r_{n-1} \qquad \text{当 } n = 1, 2, \cdots \tag{22a}$$

$$p_0 = 1, p_1 = 0.33332 \text{ 且 } p_n = \frac{4}{3} p_{n-1} - \frac{1}{3} p_{n-2} \qquad \text{当 } n = 2, 3, \cdots \tag{22b}$$

$$q_0 = 1, q_1 = 0.33332 \text{ 且 } q_n = \frac{10}{3} q_{n-1} - q_{n-2} \qquad \text{当 } n = 2, 3, \cdots \tag{22c}$$

在(22a)中,初始误差 $r_0 = 0.00004$,在(22b)和(22c)中,p_1 和 q_1 的初始误差为 $0.00001\bar{3}$。试研究每个算法的误差传播情况。

表 1.4 给出了每个序列的前 10 个数值近似解,同时表 1.5 给出了每个算法的误差。$\{r_n\}$ 的误差是稳定的,且按指数级递减。$\{p_n\}$ 的误差是稳定的。$\{q_n\}$ 的误差是不稳定的,且按指数级增长。尽管 $\{p_n\}$ 的误差是稳定的,但由于当 $p_n \to 0$ 有 $n \to \infty$,所以误差在结果中最终占支配地位,p_8 后的项无有效数字。图 1.8、图 1.9 和图 1.10 分别显示了 $\{r_n\}$,$\{p_n\}$,$\{q_n\}$ 的误差。

表 1.4　序列 $\{x_n\}=\{1/3^n\}$ 与近似值 $\{r_n\}$, $\{p_n\}$ 和 $\{q_n\}$

n	x_n	r_n	p_n	q_n
0	$1=1.0000000000$	0.9999600000	1.0000000000	1.0000000000
1	$\frac{1}{3}=0.3333333333$	0.3333200000	0.3333200000	0.3333200000
2	$\frac{1}{9}=0.1111111111$	0.1111066667	0.111093330	0.1110666667
3	$\frac{1}{27}=0.0370370370$	0.370355556	0.0370177778	0.369022222
4	$\frac{1}{81}=0.0123456790$	0.0123451852	0.0123259259	0.0119407407
5	$\frac{1}{243}=0.0041152263$	0.0041150617	0.0040953086	0.0029002469
6	$\frac{1}{729}=0.0013717421$	0.0013716872	0.0013517695	−0.0022732510
7	$\frac{1}{2187}=0.0004572474$	0.0004572291	0.0004372565	−0.0104777503
8	$\frac{1}{6561}=0.0001524158$	0.0001524097	0.0001324188	−0.0326525834
9	$\frac{1}{19\,683}=0.0000508053$	0.0000508032	0.0000308063	−0.0983641945
10	$\frac{1}{59\,049}=0.0000169351$	0.0000169344	−0.0000030646	−0.2952280648

表 1.5　误差序列 $\{x_n-r_n\}$, $\{x_n-p_n\}$ 和 $\{x_n-q_n\}$

n	x_n-r_n	x_n-p_n	x_n-q_n
0	0.0000400000	0.0000000000	0.0000000000
1	0.0000133333	0.0000133333	0.0000013333
2	0.0000044444	0.0000177778	0.0000444444
3	0.0000014815	0.0000192593	0.0001349148
4	0.0000004938	0.0000197531	0.0004049383
5	0.0000001646	0.0000199177	0.0012149794
6	0.0000000549	0.0000199726	0.0036449931
7	0.0000000183	0.0000199909	0.0109349977
8	0.0000000061	0.0000199970	0.0328049992
9	0.0000000020	0.0000199990	0.0984149998
10	0.0000000007	0.0000199997	0.2952449999

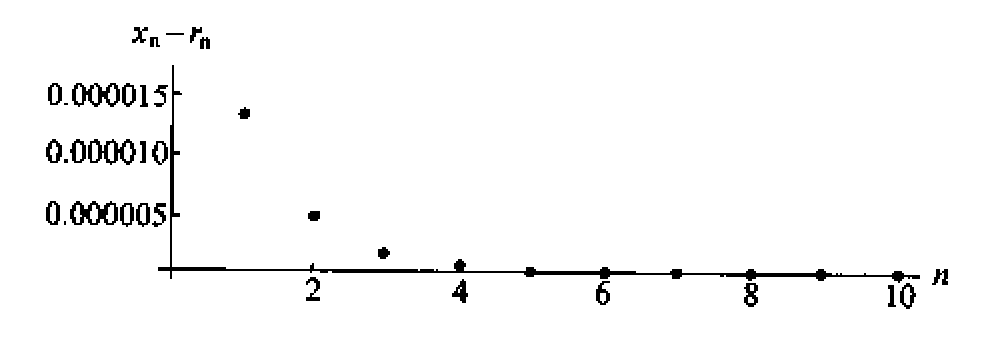


图 1.8　稳定递减的误差序列 $\{x_n-r_n\}$

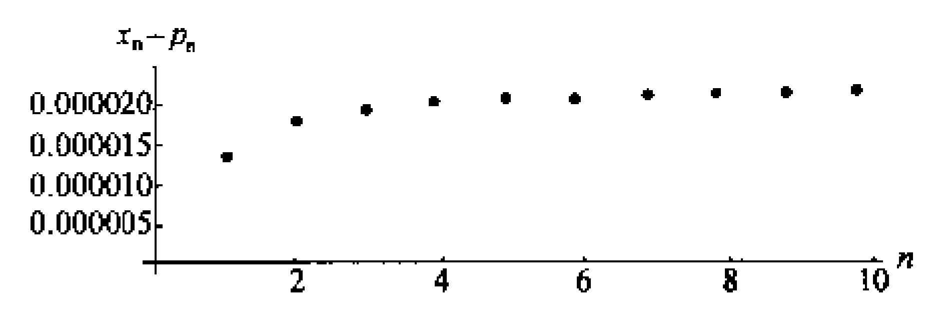


图 1.9 稳定的误差序列$\{x_n-p_n\}$

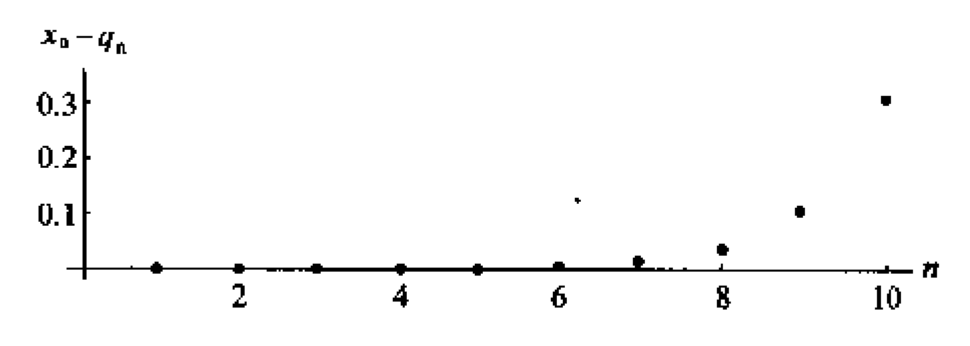


图 1.10 不稳定的误差序列$\{x_n-q_n\}$

1.3.8 数据的不确定性

从真实世界中得到的数据包含一定的不确定性和误差，这一类型的误差被称为噪声，它将影响任何数值计算的精度。采用有噪声的数据进行连续的计算并不能提高精度。因此，如果初始数据有 d 位有效数字的精度，则计算结果也只具有 d 位有效数字的精度。例如，设数据 $p_1=4.152$ 和 $p_2=0.07931$ 都有 4 位有效数字的精度，则从计算器上得出的结果($p_1+p_2=4.23131$)将被忽略。因为，从有噪声的数据得出的结果不可能比初始数据具有更多的有效数字位数，因而正确的答案应该是 $p_1+p_2=4.231$。

1.3.9 误差分析的练习

1. 求解误差 E_x 和相对误差 R_x，并判定近似值的有效数字的位数。

   (a) $x=2.71828182, \hat{x}=2.7182$

   (b) $y=98\,350, \hat{y}=98\,000$

   (c) $z=0.000068, \hat{z}=0.00006$

2. 完成下列计算：

$$\int_0^{1/4} e^{x^2}dx \approx \int_0^{1/4}\left(1+x^2+\frac{x^2}{2!}+\frac{x^6}{3!}\right)dx = \hat{p}$$

   指出在这种情况下会出现哪种类型的误差，并将计算结果与真值 $p=0.2553074606$ 进行比较。

3. (a)设 $p_1=1.414$ 和 $p_2=0.09125$，精度为 4 位有效数字，求和 p_1+p_2 与积 p_1p_2 的合适解。

   (b)设 $p_1=31.415$ 和 $p_2=0.027182$，精度为 5 位有效数字，求和 p_1+p_2 与积 p_1p_2 的合适解。

4. 完成下列计算，并指出在这种情况下会出现哪种类型的误差。

   (a) $\dfrac{\sin\left(\frac{\pi}{4}+0.00001\right)-\sin\left(\frac{\pi}{4}\right)}{0.00001}=\dfrac{0.70711385222-0.70710678119}{0.00001}=\cdots$

(b) $\frac{\ln(2+0.00005)-\ln(2)}{0.00005}=\frac{0.69317218025-0.69314718056}{0.00005}=\cdots$

5. 有时,利用三角或代数恒等式重新排列函数中的项可避免精度损失。求下列函数的等价公式以避免精度损失。

(a) $\ln(x+1)-\ln(x)$,其中 x 较大

(b) $\sqrt{x^2+1}-x$,其中 x 较大

(c) $\cos^2(x)-\sin^2(x)$,其中 $x\approx\pi/4$

(d) $\sqrt{\frac{1+\cos(x)}{2}}$,其中 $x\approx\pi$

6. 多项式求值。设 $P(x)=x^3-3x^2+3x-1$, $Q(x)=((x-3)x+3)x-1$, $R(x)=(x-1)^3$。

(a) 用 4 位舍入计算 $P(2.72)$, $Q(2.72)$, $R(2.72)$。在计算 $P(x)$时,设$(2.72)^3=20.12$, $(2.72)^2=7.398$。

(b) 用 4 位舍入计算 $P(0.975)$, $Q(0.975)$, $R(0.975)$。在计算 $P(x)$时,设$(0.975)^3=0.9268$, $(0.975)^2=0.9506$。

7. 用 3 位舍入计算下列和(按给定的顺序求和):

(a) $\sum_{k=1}^{6}\frac{1}{3^k}$ (b) $\sum_{k=1}^{6}\frac{1}{3^{7-k}}$

8. 讨论下列计算过程中的误差传播:

(a)三个数的和:

$$p+q+r=(\hat{p}+\epsilon_p)+(\hat{q}+\epsilon_q)+(\hat{r}+\epsilon_r)$$

(b)两个数的商:

$$\frac{p}{q}=\frac{\hat{p}+\epsilon_p}{\hat{q}+\epsilon_q}$$

(c)三个数的积:

$$pqr=(\hat{p}+\epsilon_p)(\hat{q}+\epsilon_q)(\hat{r}+\epsilon_r)$$

9. 设有泰勒展开式:

$$\frac{1}{1-h}=1+h+h^2+h^3+O(h^4)$$

和

$$\cos(h)=1-\frac{h^2}{2!}+\frac{h^4}{4!}+O(h^6)$$

判定它们的和与积的近似阶。

10. 设有泰勒展开式:

$$e^h=1+h+\frac{h^2}{2!}+\frac{h^3}{3!}+\frac{h^4}{4!}+O(h^5)$$

和

$$\sin(h)=h-\frac{h^3}{3!}+O(h^5)$$

判定它们的和与积的近似阶。

11. 设有泰勒展开式:

$$\cos(h)=1-\frac{h^2}{2!}+\frac{h^4}{4!}+O(h^6)$$

和

$$\sin(h) = h - \frac{h^3}{3!} + \frac{h^5}{5!} + O(h^7)$$

判定它们的和与积的近似阶。

12. 改进二次根公式。设 $a \neq 0, b^2 - 4ac > 0$，且有方程 $ax^2 + bx + c = 0$。通过如下二次根公式可解出方程的根。

    (i) $x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$ 和 $x_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$

    证明这些根可通过下列等价公式解出。

    (ii) $x_1 = \frac{-2c}{b + \sqrt{b^2 - 4ac}}$ 和 $x_2 = \frac{-2c}{b - \sqrt{b^2 - 4ac}}$

    提示：可对(i)中的分子进行有理化。

    注：当 $|b| = \sqrt{b^2 - 4ac}$ 时，必须小心处理，以避免其值过小引起巨量消失(catastrophic cancellation)，而带来精度损失。如果 $b > 0$，应该用公式(ii)计算 x_1，用公式(i)计算 x_2。如果 $b < 0$，应该用公式(i)计算 x_1，用公式(ii)计算 x_2。

13. 利用练习 12 中求解 x_1 和 x_2 的适当公式，计算下列二次方程的根。

    (a) $x^2 - 1\,000.001x + 1 = 0$

    (b) $x^2 - 10\,000.0001x + 1 = 0$

    (c) $x^2 - 100\,000.00001x + 1 = 0$

    (d) $x^2 - 1\,000\,000.000001x + 1 = 0$

1.3.10 算法和程序

1. 根据练习 12 和练习 13 构造算法和 MATLAB 程序，以便精确计算所有情况下的二次方程的根，包括 $|b| \approx \sqrt{b^2 - 4ac}$ 的情况。

2. 参照例 1.25，对下列 3 个差分方程计算出前 10 个数值近似值。在每种情况下引入一个小初始误差。如果没有初始误差，则每个差分方程将生成序列 $\{1/2^n\}_{n=1}^{\infty}$。构造类似表 1.4、表 1.5 和图 1.8、图 1.9、图 1.10 的输出。

    (a) $r_0 = 0.994,\ r_n = \frac{1}{2} r_{n-1}$，其中 $n = 1, 2, \cdots$

    (b) $p_0 = 1, p_1 = 0.497, p_n = \frac{3}{2} p_{n-1} - p_{n-2}$，其中 $n = 2, 3, \cdots$

    (c) $q_0 = 1, q_1 = 0.497,\ q_n = \frac{5}{2} q_{n-1} - q_{n-2}$，其中 $n = 2, 4, \cdots$

第2章　非线性方程 $f(x)=0$ 的解法

考虑一个涉及球体的物理问题,球体的半径为 r,并浸没入水中,深度为 d(如图2.1所示)。假设这个球由一种密度 $\rho=0.638$ 的长叶松构成,且它的半径 $r=10$ cm。当球浸入水中时,它浸没在水中的质量为多少?

当一个球体以深度 d 浸没入水中时,所排开水的质量 M_w 为:

$$M_w=\int_0^d \pi(r^2-(x-r)^2)dx=\frac{\pi d^2(3r-d)}{3}$$

而球的质量 $M_b=4\pi r^3\rho/3$。根据阿基米德(Archimedes)定律,有 $M_w=M_b$,则产生需要求解的方程如下:

$$\frac{\pi(d^3-3d^2r+4r^3\rho)}{3}=0$$

在下述例子中($r=10,\rho=0.638$)方程变为:

$$\frac{\pi(2552-30d^2+d^3)}{3}=0$$

三次多项式 $y=2552-30d^2+d^3$ 的形状如图2.2所示,而且根据此图,可发现解在 $d=12$ 附近。

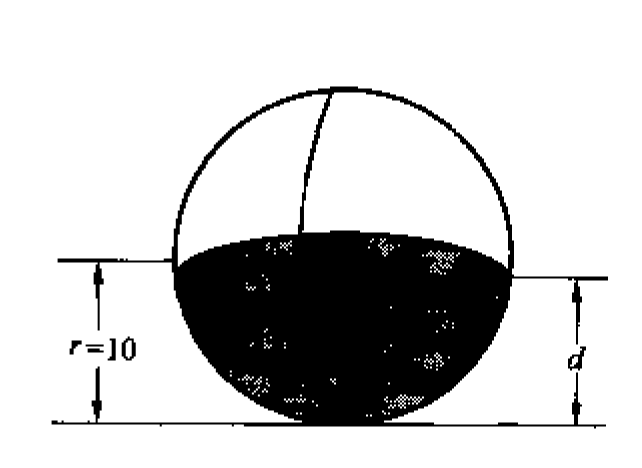


图2.1　浸入水中深度为 d、半径为 r 的球体部分

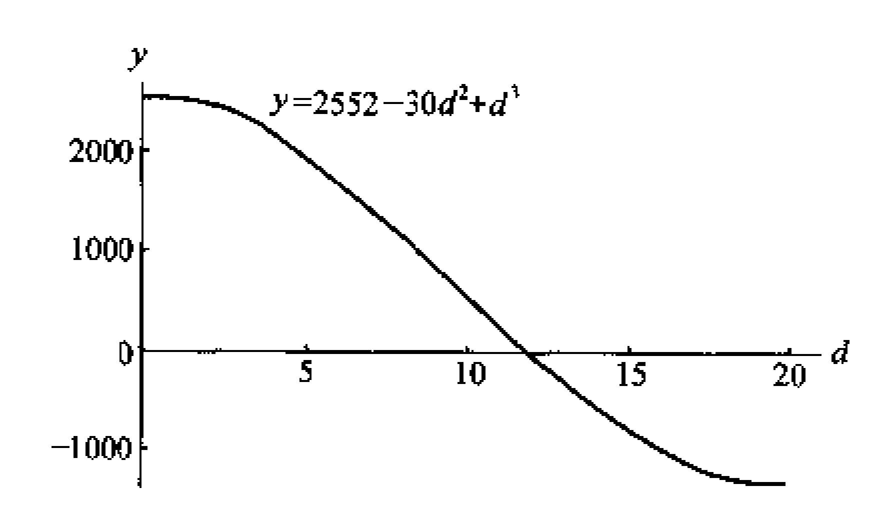


图2.2　三次多项式 $y=2552-30d^2+d^3$

这一章的目标是研究求解方程根的各种方法。例如,采用对分法可得到上述方程的三个根 $d_1=-8.17607212$、$d_2=11.86150151$ 和 $d_3=26.31457061$。第一个根不是此问题的可行解,因为它是负数。第三个根大于球体直径,也不是需要的解。根 $d_2=11.86150151$ 位于区间[0,20]内,是合适的解。它的大小是合适的,因为球体的一大半一定是浸入水中的。

2.1　求解 $x=g(x)$ 的迭代法

计算机科学中的一个基本要素是迭代(iteration)。正如名字所表示的含义,迭代是指重复执行一个计算过程,直到找到答案。迭代技术用来求解方程的根、线性和非线性方程组的解以

及微分方程的解。这一节主要研究重复替换的迭代处理过程。

首先需要有一个用于逐项计算的规则或函数 $g(x)$，并且有一个起始点 p_0。然后通过迭代规则 $p_{k+1}=g(p_k)$，可得到序列值 $\{p_k\}$。此序列有如下模式(其中 p_0 为初始值)：

$$\begin{aligned} &p_0 \\ p_1 &= g(p_0) \\ p_2 &= g(p_1) \\ &\vdots \\ p_k &= g(p_{k-1}) \\ p_{k+1} &= g(p_k) \\ &\vdots \end{aligned} \tag{1}$$

从一个数的无限序列可得到什么呢？如果这些数趋向一个极限，则求解目的达到。但如果这些数发散或周期性重复呢？下面的例子给出了这种情况。

例 2.1　迭代规则为 $p_0=1$，且 $p_{k+1}=1.001p_k$，其中 $k=0,1,\ldots$。它产生一个发散序列，前 100 项为：

$$\begin{aligned} p_1 &= 1.001p_0 = (1.001)(1.000000) = 1.001000 \\ p_2 &= 1.001p_1 = (1.001)(1.001000) = 1.002001 \\ p_3 &= 1.001p_2 = (1.001)(1.002001) = 1.003003 \\ &\vdots \\ p_{100} &= 1.001p_{99} = (1.001)(1.104012) = 1.105116 \end{aligned}$$

这个过程可无限持续下去，而且可以很容易看出 $\lim_{n\to\infty} p_n=+\infty$。在第 9 章中，将看到序列 $\{p_k\}$ 是微分方程 $y'=0.001y$ 的数值解。这个解为 $y(x)=e^{0.001x}$。实际上，如果比较序列中的第 100 项和 $y(100)$，可以发现 $p_{100}=1.105116\approx1.105171=e^{0.1}=y(100)$。

这一节中，主要关注产生收敛序列 $\{p_k\}$ 的函数 $g(x)$ 的类型。

2.1.1　寻求固定点

定义 2.1　(固定点)函数 $g(x)$ 的一个固定点是指一个实数 P，满足 $P=g(P)$。

从图形角度分析，函数 $g(x)$ 的固定点是 $y=g(x)$ 和 $y=x$ 的交点。

定义 2.2　(固定点迭代)迭代 $p_{n+1}=g(p_n)$，其中 $n=0,1,\cdots$，称为固定点迭代。

定理 2.1　设 g 是一连续函数，且 $\{p_n\}_{n=0}^{\infty}$ 是由固定点迭代生成的序列。如果 $\lim_{n\to\infty} p_n=P$，则 P 是 $g(x)$ 的固定点。

证明：如果 $\lim_{n\to\infty} p_n=P$，则 $\lim_{n\to\infty} p_{n+1}=P$。根据这个结论，$g$ 的连续性和 $p_{n+1}=g(p_n)$ 存在如下关系：

$$g(P)=g(\lim_{n\to\infty}p_n)=\lim_{n\to\infty}g(p_n)=\lim_{n\to\infty}p_{n+1}=P \tag{2}$$

因此，P 是 $g(x)$ 的固定点。

例 2.2 设有收敛迭代

$$p_0 = 0.5,\ p_{k+1} = e^{-p_k},\ k = 0, 1, \cdots$$

前十项的计算结果如下所示:

$$\begin{aligned} p_1 &= e^{-0.500000} = 0.606531 \\ p_2 &= e^{-0.606531} = 0.545239 \\ p_3 &= e^{-0.545239} = 0.579703 \\ &\vdots \qquad\quad \vdots \\ p_9 &= e^{-0.566409} = 0.567560 \\ p_{10} &= e^{-0.567560} = 0.566907 \end{aligned}$$

这个序列是收敛的,且进一步计算可发现:

$$\lim_{n\to\infty} p_n = 0.567143\cdots$$

这样,可找到函数 $y = e^{-x}$ 的固定点近似值。

下列两个定理建立了固定点存在性条件,以及寻找固定点迭代过程的收敛性条件。

定理 2.2 设函数 $g \in C[a,b]$。

如果对于所有的 $x \in [a,b]$,映射 $y = g(x)$ 的范围满足 $y \in [a,b]$,则函数 g 在 $[a,b]$ 内有一个固定点。 (3)

此外,设 $g'(x)$ 定义在 (a,b) 内,且对于所有的 $x \in (a,b)$,存在正常数 $K < 1$,使得 $|g'(x)| \leqslant K < 1$,则函数 g 在 $[a,b]$ 内有惟一的固定点 P。 (4)

对命题(3)的证明:如果 $g(a) = a$ 或 $g(b) = b$,则断言为真。否则,$g(a)$ 必须满足 $g(a) \in [a,b]$,$g(b)$ 的值必须满足 $g(b) \in [a,b]$。表达式 $f(x) \equiv x - g(x)$ 有如下特性:

$$f(a) = a - g(a) < 0 \text{ 且 } f(b) = b - g(b) > 0$$

对 $f(x)$ 应用定理 1.2(中值定理),而且由于常量 $L = 0$,可推断出存在数 P,且 $P \in (a,b)$,满足 $f(P) = 0$。因此,$P = g(P)$,且 P 是 $g(x)$ 的固定点。

对命题(4)的证明:必须证明结果是惟一的。采用反证法,设存在两个固定点 P_1 和 P_2。根据定理 1.6(均值定理),可推断出存在数 $d \in (a,b)$,满足:

$$g'(d) = \frac{g(P_2) - g(P_1)}{P_2 - P_1} \tag{5}$$

根据假设有 $g(P_1) = P_1$ 且 $g(P_2) = P_2$,并对等式(5)的右边进行简化可得:

$$g'(d) = \frac{P_2 - P_1}{P_2 - P_1} = 1$$

但这与(4)中的假设在 (a,b) 内有 $|g'(x)| < 1$ 矛盾,因此不可能存在两个固定点。所以,在命题(4)的假设条件下,$g(x)$ 在 $[a,b]$ 内有一个惟一的固定点 P。

例 2.3 根据定理 2.2 严格地证明 $g(x) = \cos(x)$ 在 $[0,1]$ 内有一个惟一的固定点。

显然,$g \in C[0,1]$。其次,$g(x) = \cos(x)$ 在 $[0,1]$ 内是递减函数,因此它在 $[0,1]$ 内的范围是 $[\cos(1), 1] \subseteq [0,1]$。这样可满足定理 2.2 的条件(3),且 g 在 $[0,1]$ 内有一固定点。最后,如果 $x \in (0,1)$,则 $|g'(x)| = |-\sin(x)| = \sin(x) \leqslant \sin(1) < 0.8415 < 1$。这样 $K = \sin(1) <$

1，可满足定理 2.2 的命题(4)，所以 g 在$[0,1]$内有惟一的固定点。

现在可指定一个定理来判断(1)中给出的固定点迭代过程算法是否将产生一个收敛序列或发散序列。

定理 2.3(固定点定理)　设有(i)$g,g'\in C[a,b]$，(ii)K 是一个正常数，(iii)$p_0\in(a,b)$，(iv)对所有 $x\in[a,b]$，有 $g(x)\in[a,b]$。

如果对所有 $x\in[a,b]$有$|g'(x)|\leqslant K<1$，则迭代 $P_n=g(P_{n-1})$将收敛到一个惟一固定点 $P\in[a,b]$。在这种情况下，P 称为吸引(attractive)固定点。 (6)

如果对所有 $x\in[a,b]$有$|g'(x)|>1$，则迭代 $P_n=g(P_{n-1})$将不会收敛到 P。在这种情况下，P 称为排斥(repelling)固定点，而且迭代显示出局部发散性。 (7)

注 1：在命题(7)中假设 $p_0\neq P$。

注 2：因为函数 g 在包含 P 的一段间隔中是连续的，可在命题(6)和命题(7)中分别利用更简单的判别条件$|g'(P)|\leqslant K<1$和$|g'(P)|>1$。

证明：首先要证明点$\{p_n\}_{n=0}^{\infty}$都位于(a,b)内。从 p_0 开始，根据定理 1.6(均值定理)，可推导出存在一个值 $c_0\in(a,b)$满足：

$$\begin{aligned}|P-p_1|&=|g(P)-g(p_0)|=|g'(c_0)(P-p_0)|\\&=|g'(c_0)||P-p_0|\leqslant K|P-p_0|<|P-p_0|\end{aligned}\tag{8}$$

因此，p_1 比 p_0 更接近 P，且 $p_1\in(a,b)$(参看图 2.3)。一般情况下，设 $p_{n-1}\in(a,b)$，则：

$$\begin{aligned}|P-p_n|&=|g(P)-g(p_{n-1})|=|g'(c_{n-1})(P-p_{n-1})|\\&=|g'(c_{n-1})||P-p_{n-1}|\leqslant K|P-p_{n-1}|<|P-p_{n-1}|\end{aligned}\tag{9}$$

因此，$p_n\in(a,b)$，而且可归纳出所有的点$\{p_n\}_{n=0}^{\infty}$位于(a,b)内。

图 2.3　$P,p_0,p_1,|P-p_0|$和$|P-p_1|$之间的关系

为完成(6)的证明，应证明如下表达式成立：

$$\lim_{n\to\infty}|P-p_n|=0\tag{10}$$

首先，用归纳法的证明可建立如下不等式：

$$|P-p_n|\leqslant K^n|P-p_0|\tag{11}$$

$n=1$ 时满足关系(8)。利用归纳假设$|P-p_{n-1}|\leqslant K^{n-1}|P-p_0|$和式(9)，可得到：

$$|P-p_n|\leqslant K|P-p_{n-1}|\leqslant KK^{n-1}|P-p_0|=K^n|P-p_0|$$

这样，通过归纳法得出，对所有的 n 满足不等式(11)。由于 $0<K<1$，所以当 n 趋于无穷大时，项 K^n 趋近于 0。因此：

$$0\leqslant\lim_{n\to\infty}|P-p_n|\leqslant\lim_{n\to\infty}K^n|P-p_0|=0\tag{12}$$

$|P-p_n|$的极限压缩在 0 的左边和 0 的右边之间，所以，可得出$\lim_{n\to\infty}|P-p_n|=0$。这样 $\lim_{n\to\infty}p_n=P$，且根据定理 2.1，迭代 $p_n=g(p_{n-1})$收敛到固定点 P。因此定理 2.3 的命题(6)得证。读者可自行研究命题(7)。

推论 2.1　设函数 g 满足定理 2.3 中(6)给出的假设。当用 p_n 近似表示 P 时,引入的误差的边界如下所示:

对所有的 $n \geqslant 1$ 有　　$|P - p_n| \leqslant K^n |P - p_0|$　　(13)

且对所有的 $n \geqslant 1$ 有　　$|P - p_n| \leqslant \dfrac{K^n |p_1 - p_0|}{1 - K}$　　(14)

2.1.2　固定点迭代的图形解释

由于需要寻找 $g(x)$ 的固定点 P,曲线 $y = g(x)$ 和直线 $y = x$ 必须相交在点 (P, P)。两种类型的收敛迭代:单调收敛迭代和振荡收敛迭代分别如图 2.4(a)和图 2.4(b)所示。

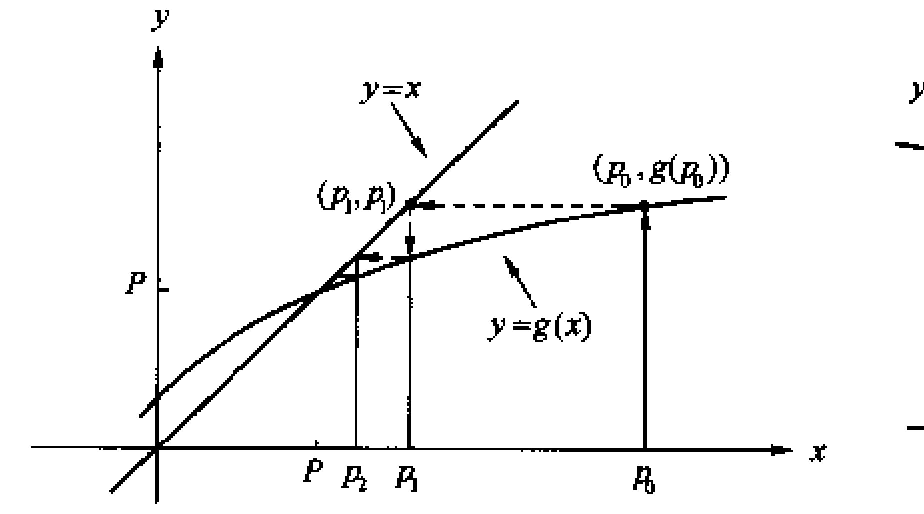
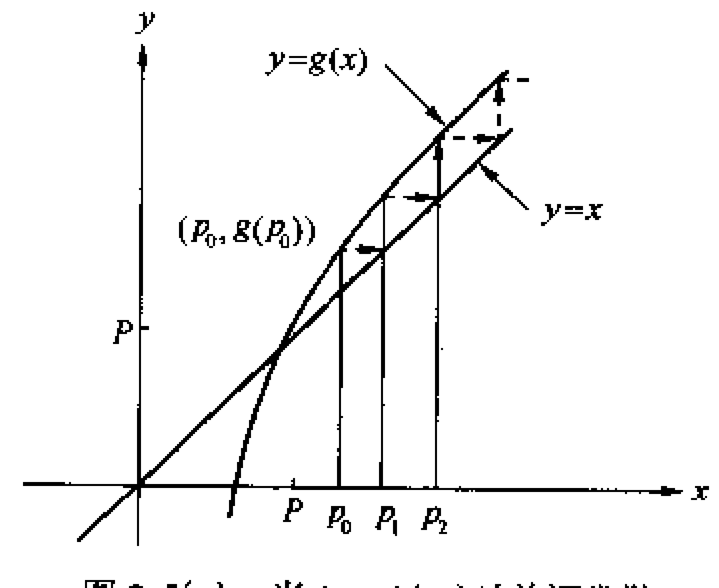


图 2.4(a)　当 $0 < g'(P) < 1$ 时单调收敛　　图 2.4(b)　当 $-1 < g'(P) < 0$ 时振荡收敛

为了直观地描述迭代过程,从 x 轴的 p_0 开始,首先纵向移动到曲线 $y = g(x)$ 上的点 $(p_0, p_1) = (p_0, g(p_0))$。然后从 (p_0, p_1) 横向移动到直线 $y = x$ 上的点 (p_1, p_1)。最后,纵向向下移动到 x 轴上的 p_1。利用递归式 $p_{n+1} = g(p_n)$ 构造图中的点 (p_n, p_{n+1}),然后横向移动定位到直线 $y = x$ 上的点 (p_{n+1}, p_{n+1}),接着纵向移动到 x 轴上的点 p_{n+1}。整个过程如图 2.4 所示。

如果 $|g'(P)| > 1$,则迭代 $p_{n+1} = g(p_n)$ 产生的序列对 P 发散。两种简单类型的发散迭代:单调发散迭代和振荡发散迭代分别如图 2.5(a)和图 2.5(b)所示。



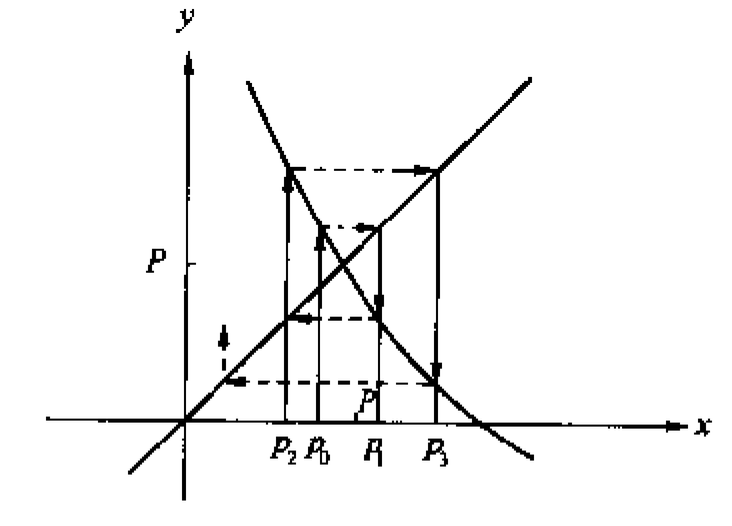


图 2.5(a)　当 $1 < g'(P)$ 时单调发散　　图 2.5(b)　当 $g'(P) < -1$ 时振荡发散

例 2.4　当使用函数 $p_{n+1} = g(p_n)$ 时,设迭代 $p_{n+1} = g(p_n)$。通过求解方程 $x = g(x)$ 可找到固定点。两个解(函数 g 的固定点)分别为 $x = -2$ 和 $x = 2$。函数的导数是 $g'(x) = 1 - x/2$,这里只考虑两种情况:

情况(i)：　$P=-2$

从　$p_0=-2.05$　开始

然后得到　$p_1=-2.100625$

$p_2=-2.20378135$

$p_3=-2.41794441$

$\vdots$

$\lim_{n\to\infty} p_n=-\infty$

因为 $|g'(x)|>\frac{3}{2}$ 在 $[-3,-1]$ 上，根据定理 2.3，序列不收敛到 $P=-2$。

情况(ii)：　$P=2$

从　$p_0=1.6$　开始

然后得到　$p_1=1.96$

$p_2=1.9996$

$p_3=1.99999996$

$\vdots$

$\lim_{n\to\infty} p_n=2$

因为 $[1,3]$ 上 $|g'(x)|<\frac{1}{2}$，根据定理 2.3，序列收敛到 $P=2$。

定理 2.3 并没有指出当 $g'(P)=1$ 时将发生什么情况。下面的例子专门构造出这种情况，这样只要 $p_0>P$，则序列 $\{p_n\}$ 收敛，同时如果 $p_0<P$，则序列发散。

例 2.5　当使用函数 $g(x)=2(x-1)^{1/2}$ 且 $x\geqslant 1$ 时，设迭代 $p_{n+1}=g(p_n)$。这样只有一个固定点 $P=2$ 存在。函数的导数为 $g'(x)=1/(x-1)^{1/2}$，且 $g'(2)=1$，因此不能应用定理 2.3。当初始值位于点 $P=2$ 的左边和右边时的两种情况如下所示。

情况(i)：从　$p_0=1.5$ 开始

然后得到　$p_1=1.41421356$

$p_2=1.28718851$

$p_3=1.07179943$

$p_4=0.53590832$

$\vdots$

$p_5=2(-0.46409168)^{1/2}$

因为 P_4 在 $g(x)$ 的域外，不能计算项 P_5。

情况(ii)：从　$p_0=2.5$ 开始

然后得到　$p_1=2.44948974$

$p_2=2.40789513$

$p_3=2.37309514$

$p_4=2.34358284$

$\vdots$

$\lim_{n\to\infty} p_n=2$

这个序列收敛到 $P=2$ 太慢，实际上 $P_{1000}=2.00398714$。

2.1.3　绝对误差和相对误差

在例 2.5 中，序列收敛很慢，1000 次迭代后三个连续项为：

$$p_{1000}=2.00398714,\quad p_{1001}=2.00398317,\quad \text{和}\quad p_{1002}=2.00397921$$

这不会产生混淆，因为可以通过计算更多的项寻找到更好的近似值！但中止迭代的判别条件是什么呢？如果注意到连续项的差异：

$$|p_{1001}-p_{1002}|=|2.00398317-2.00397921|=0.00000396$$

然而近似值 p_{1000} 的绝对误差是：

$$|P-p_{1000}|=|2.00000000-2.00398714|=0.00398714$$

这比 $|p_{1001}-p_{1002}|$ 大 1000 倍，这种情况说明了连续项的相近并不能保证精度。但通常连续项的差异比较是中止迭代过程的惟一判别条件。

程序 2.1(固定点迭代) 求解方程 $x=g(x)$的近似值,初始值为 p_0,迭代式为 $p_{n+1}=g(p_n)$

```
function [k,p,err,P] = fixpt(g,p0,tol,max1)
% Input  - g is the iteration function input as a string 'g'
%        - p0 is the initial guess for the fixed point
%        - tol is the tolerance
%        - max1 is the maximum number of iterations
% Output- k is the number of iterations that were carried out
%        - p is the approximation to the fixed point
%        - err is the error in the approximation
%        - P contains the sequence {pn}
P(1) = p0;
for k = 2:max1
    P(k) = feval (g,P(k-1));
    err = abs(P(k) - P(k-1));
    relerr = err/(abs(P(k)) + eps);
    p = P(k);
    if (err< tol) | (relerr < tol),break;end
end
if k = = max1
    disp('maximum number of iterations exceeded')
end
P = P';
```

注:当使用用户定义的函数 fixpt 时,必须输入 M 文件 g.m 作为字符串'g'(参见附录"MATLAB 介绍")。

2.1.4 求解 $x=g(x)$迭代过程的练习

1. 在给定的区间间隔内,判定下列每个函数是否有惟一的固定点(参照例 2.3)。
   (a)$g(x)=1-x^2/4$ 在区间[0,1]内
   (b)$g(x)=2^{-x}$在区间[0,1]内
   (c)$g(x)=1/x$ 在区间[0.5,5.2]内
2. 当

$$g(x)=-4+4x-\frac{1}{2}x^2$$

   时,研究固定点迭代的性质。
   (a)求解 $g(x)=x$,且证明 $P=2$ 和 $P=4$ 是固定点。
   (b)用初始值 $p_0=1.9$,计算 p_1,p_2 和 p_3。
   (c)用初始值 $p_0=3.8$,计算 p_1,p_2 和 p_3。
   (d)对在(b)和(c)中的 p_k,寻找误差 E_k 和相对误差 R_k。
   (e)从定理 2.3 中可得出什么结论?
3. 在同一坐标内对 $g(x)$、直线 $y=x$ 和给定的固定点 P 画图。使用给定的初始值 p_0,计算 p_1 和 p_2。构造如图 2.4 和图 2.5 的图形。根据构造的图形,从图形的角度判断固定点迭代是否收敛。

(a) $g(x)=(6+x)^{1/2}$, $P=3$ 和 $p_0=7$

(b) $g(x)=1+2/x$, $P=2$ 和 $p_0=4$

(c) $g(x)=x^2/3$, $P=3$ 和 $p_0=3.5$

(d) $g(x)=-x^2+2x+2$, $P=2$ 和 $p_0=2.5$

4. 设 $g(x)=x^2+x-4$,能否利用固定点迭代求解方程 $x=g(x)$? 为什么?
5. 设 $g(x)=x\cos(x)$。求解 $x=g(x)$,且寻找函数 g 的所有固定点(有有限个)。能否利用固定点迭代求解方程 $x=g(x)$? 为什么?
6. 设 $g(x)$和 $g'(x)$在区间(a,b)上有定义且连续;并且 $p_0,p_1,p_2\in(a,b)$;而且 $p_1=g(p_0)$, $p_2=g(p_1)$。假设存在常量 K 满足$|g'(x)|<K$,证明$|p_2-p_1|<K|p_1-p_0|$。提示:利用均值定理。
7. 设 $g(x)$和 $g'(x)$在区间(a,b)上连续,且在此区间内$|g'(x)|>1$。如果固定点 P 和初始近似值 p_0,p_1 位于区间(a,b)内,试证明 $p_1=g(p_0)$,即$|E_1|=|P-p_1|>|P-p_0|=|E_0|$,因此可建立定理 2.3 中的命题(7)(局部发散)。
8. 设 $g(x)=-0.0001x^2+x$,且 $p_0=1$,考虑固定点迭代。

   (a)证明 $p_0>p_1>\cdots>p_n>p_{n+1}>\cdots$。

   (b)证明对所有 n,有 $p_n>0$。

   (c)由于序列$\{p_n\}$递减,有下界,所以它有一极限。请问极限是什么?
9. 设 $g(x)=0.5x+1.5$,且 $p_0=4$,考虑固定点迭代。

   (a)证明固定点为 $P=3$。

   (b)证明$|P-p_n|=|P-p_{n-1}|/2$,其中 $n=1,2,3,\cdots$。

   (c)证明$|P-p_n|=|P-p_0|/2^n$,其中 $n=1,2,3,\cdots$。
10. 设 $g(x)=x/2$,考虑固定点迭代。

    (a)求值$|p_{k+1}-p_k|/|p_{k+1}|$。

    (b)如果只利用程序 2.1 中的相对误差作为停止判别的条件,将发生什么情况?
11. 为什么当 $g'(P)\approx 0$ 时,对于固定点迭代过程有好处?

2.1.5 算法和程序

1. 使用程序 2.1 求解下面每个函数的固定点(尽可能多)近似值,答案精确到小数点后 12 位。同时,构造每个函数和直线 $y=x$ 的图来显示所有的固定点。

   (a) $g(x)=x^5-3x^3-2x^2+2$

   (b) $g(x)=\cos(\sin(x))$

   (c) $g(x)=x^2-\sin(x+0.15)$

   (d) $g(x)=x^{x-\cos(x)}$

2.2 定位一个根的划分方法(bracketing methods)

考虑一个与利息相关的题目。假设每个月存钱 P,且年利率为 I。存了 N 次后,钱的总数是:

$$A = P + P\left(1+\frac{I}{12}\right) + P\left(1+\frac{I}{12}\right)^2 + \cdots + P\left(1+\frac{I}{12}\right)^{N-1} \tag{1}$$

方程右边的第一项是最近的钱数。得到一次利息的第一次报酬是 $P\left(1+\frac{I}{12}\right)$。得到两次利息的第二次报酬是 $P\left(1+\frac{I}{12}\right)^2$,等等。最后,得到 $N-1$ 次利息的最近的报酬是 $P\left(1+\frac{I}{12}\right)^{N-1}$。求解 N 项几何级数和的公式是:

$$1 + r + r^2 + r^3 + \cdots + r^{N-1} = \frac{1-r^N}{1-r} \tag{2}$$

可将式(1)写成如下形式:

$$A = P\left(1+\left(1+\frac{I}{12}\right)+\left(1+\frac{I}{12}\right)^2+\cdots+\left(1+\frac{I}{12}\right)^{N-1}\right)$$

而且在(2)中用 $r=(1+I/12)$进行替换可得:

$$A = P\,\frac{1-\left(1+\frac{I}{12}\right)^N}{1-\left(1+\frac{I}{12}\right)}$$

这可简化得到应付年金的公式:

$$A = \frac{P}{I/12}\left(\left(1+\frac{I}{12}\right)^N - 1\right) \tag{3}$$

下面的例子使用应付年金的公式而且需要一系列的重复计算来得到答案。

例 2.6 *每个月存 \$250,并持续 20 年,希望在 20 年后报酬和利息的总值达到 \$250 000。利率 I 为多少时可满足需求?*

如果 $N=240$,则 A 只是 I 的函数,即 $A=A(I)$。起始假设 $I_0=0.12$ 和 $I_1=0.13$,执行一系列的计算来接近最终答案。从 $I_0=0.12$ 开始可得:

$$A(0.12) = \frac{250}{0.12/12}\left(\left(1+\frac{0.12}{12}\right)^{240}-1\right) = 247\,314$$

由于此结果比目标小,接下来试验 $I_1=0.13$,计算如下:

$$A(0.13) = \frac{250}{0.13/12}\left(\left(1+\frac{0.13}{12}\right)^{240}-1\right) = 282\,311$$

结果又有些高,因此取中间值 $I_2=0.125$,计算如下:

$$A(0.125) = \frac{250}{0.125/12}\left(\left(1+\frac{0.125}{12}\right)^{240}-1\right) = 264\,623$$

这个结果还有点高,这样可得出期望的利率在区间[0.12,0.125]内。下一个猜想值是中间点 $I_3=0.1225$,计算如下:

$$A(0.1225) = \frac{250}{0.1225/12}\left(\left(1+\frac{0.1225}{12}\right)^{240}-1\right) = 255\,803$$

这个结果还是有点高,并且区间压缩到[0.12,0.1225]内。最后使用中间点 $I_4=0.12125$ 进行计算,计算如下:

$$A(0.12125) = \frac{250}{0.12125/12}\left(\left(1+\frac{0.12125}{12}\right)^{240}-1\right) = 251\,518$$

如果需要更多的有效位数，可进行进一步的迭代。这个例子的目的是对特定的 L 寻找 I，使得 $A(I)=L$。将常量 L 放在左边并求解 $A(I)-L=0$ 是一个标准的方法。

定义 2.3（方程的根，函数的零点） 设 $f(x)$ 是连续函数。满足 $f(r)=0$ 的任意 r 成为方程 $f(x)=0$ 的一个根。也称 r 为函数 $f(x)$ 的零点。

例如，方程 $2x^2+5x-3=0$ 有两个实根 $r_1=0.5$ 和 $r_2=-3$，而且对应的函数 $f(x)=2x^2+5x-3=(2x-1)(x+3)$ 有两个实零点 $r_1=0.5$ 和 $r_2=-3$。

2.2.1 波尔察诺(Bolzano)二分法

这一节将开发第一个划分方法来寻找连续函数的零点。起始区间 $[a,b]$ 必须满足 $f(a)$ 与 $f(b)$ 符号相反的条件。由于连续函数 $y=f(x)$ 的图形无间断，所以它会在零点 $x=r$ 处跨过 x 轴，且 r 在区间内（如图 2.6 所示）。通过二分法可将区间内的端点逐步逼近零点，直到得到一个任意小的包含零点的间隔。二分法判定过程的第一步是选择中点 $c=(a+b)/2$，然后分析可能存在的三种情况：

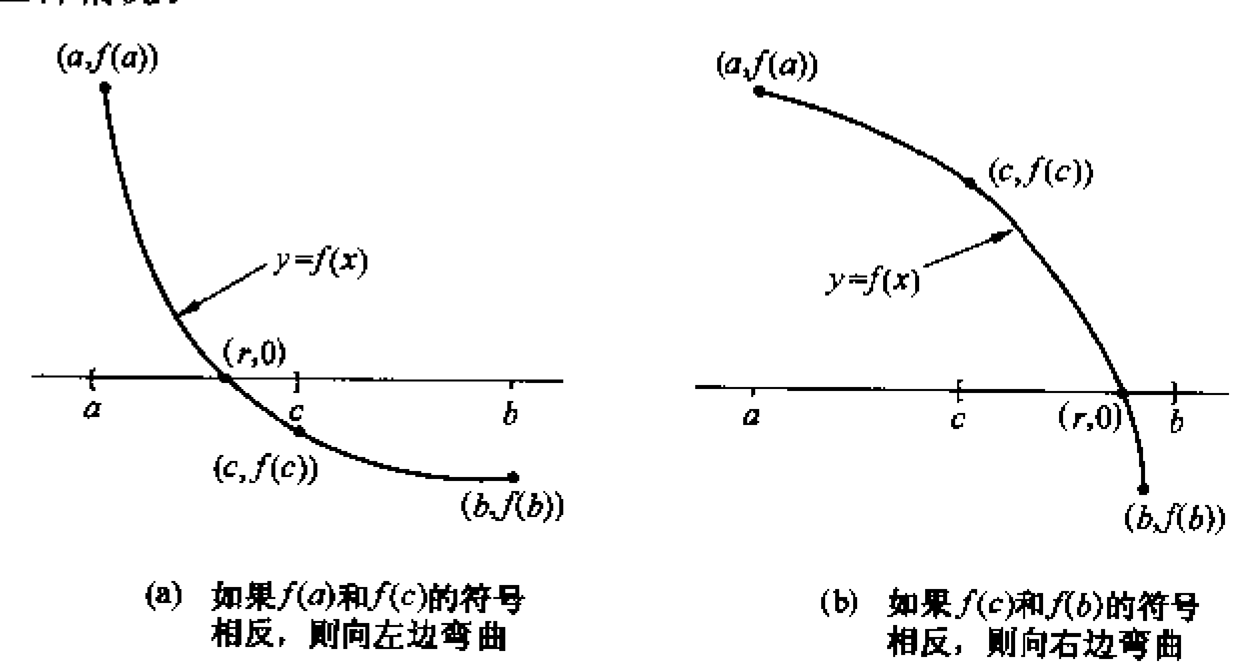


图 2.6　二分法的判定过程

如果 $f(a)$ 和 $f(c)$ 符号相反，则在区间 $[a,c]$ 内存在零点。 (4)

如果 $f(c)$ 和 $f(b)$ 符号相反，则在区间 $[c,b]$ 内存在零点。 (5)

如果 $f(c)=0$，则 c 是零点。 (6)

如果情况(4)或(5)发生，则表示找到一个比原先区间范围小一半的区间，它包含根，并称为对区间进行压缩（如图 2.6 所示）。为了持续此过程，需要对新的更小区间 $[a,b]$ 进行重新标号，重复执行直到区间足够小。由于二分法过程包括嵌套区间间隔和它们的中点，所以采用如下符号来表示过程的细节：

$[a_0,b_0]$ 是起始区间，$c_0=\dfrac{a_0+b_0}{2}$ 是中点

$[a_1,b_1]$ 是第二个区间，它包含零点 r，同时 c_1 是中点

区间 $[a_1,b_1]$ 的宽度范围是 $[a_0,b_0]$ 的一半 (7)

$[a_{n+1},b_{n+1}]$ 得到第 n 个区间 $[a_n,b_n]$（包含 r，并有中点 c_n）后，

可构造出 $[a_{n+1},b_{n+1}]$，它也包括 r，宽度范围是 $[a_n,b_n]$ 的一半。

留一个练习给读者:如何证明左端点是递增的,右端点是递减的,即:

$$a_0 \leqslant a_1 \leqslant \cdots \leqslant a_n \leqslant \cdots \leqslant r \leqslant \cdots \leqslant b_n \leqslant \cdots \leqslant b_1 \leqslant b_0 \tag{8}$$

这里 $c_n = \frac{a_n + b_n}{2}$,且如果 $f(a_{n+1}f(b_{n+1})) < 0$,则:

对所有的 n,有

$$[a_{n+1}, b_{n+1}] = [a_n, c_n] \text{或} [a_{n+1}, b_{n+1}] = [c_n, b_n] \tag{9}$$

定理 2.4(二分法定理)　设 $f \in C[a,b]$,且存在数 $r \in [a,b]$ 满足 $f(r) = 0$。如果 $f(a)$ 和 $f(b)$ 的符号相反,且 $\{c_n\}_{n=0}^{\infty}$ 表示式(8)和式(9)中二分法生成的中点序列,则:

$$|r - c_n| \leqslant \frac{b-a}{2^{n+1}} \qquad \text{其中 } n = 0,1,\cdots, \tag{10}$$

这样序列 $\{c_n\}_{n=0}^{\infty}$ 收敛到零点 $x = r$,即可表示为:

$$\lim_{n\to\infty} c_n = r \tag{11}$$

证明:由于零点 r 和中点 c_n 都位于区间 $[a_n, b_n]$ 内,c_n 与 r 之间的距离不会比这个区间的一半宽度范围大(如图 2.7 所示)。这样:

对所有 n 有:

$$|r - c_n| \leqslant \frac{b_n - a_n}{2} \tag{12}$$

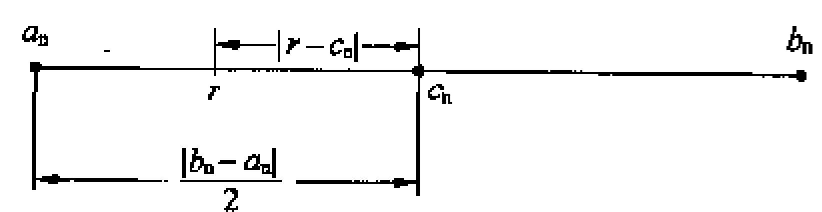


图 2.7　用于二分法的根 r 和区间 $[a_n, b_n]$ 内的中点 c_n

观察连续的区间宽度范围,可得到如下模式:

$$b_1 - a_1 = \frac{b_0 - a_0}{2^1}$$

$$b_2 - a_2 = \frac{b_1 - a_1}{2} = \frac{b_0 - a_0}{2^2}$$

留一个练习给读者:使用数学归纳法证明:

$$b_n - a_n = \frac{b_0 - a_0}{2^n} \tag{13}$$

结合式(12)和式(13)可得到对所有 n 有:

$$|r - c_n| \leqslant \frac{b_0 - a_0}{2^{n+1}} \tag{14}$$

现在利用定理 2.3 中的一个类似论点来证明式(14)意味着序列 $\{c_n\}_{n=0}^{\infty}$ 收敛到 r,这样定理得证。

例 2.7　在无阻尼强迫振荡的研究中会碰到函数 $h(x) = x\sin(x)$。寻找在区间[0,2]内的值 x,满足 $h(x) = 1$(函数 $\sin(x)$ 用弧度计算)。

利用二分法寻找函数 $f(x)=x\sin(x)-1$ 的零点。初始值 $a_0=0, b_0=2$。计算：

$$f(0)=-1.000000 \quad 和 \quad f(2)=0.818595,$$

因此 $f(x)=0$ 的一个根位于 $[0,2]$ 内。在中点 $c_0=1$，可发现 $f(1)=-0.158529$。因此区间改变为 $[c_0,b_0]=[1,2]$。

接下来，从左边压缩使得 $a_1=c_0$ 且 $b_1=b_0$。中点 $c_1=1.5$ 且 $f(c_1)=0.496242$。现在 $f(1)=-0.158529$ 且 $f(1.5)=0.496242$，这表示根位于区间 $[a_1,c_1]=[1.0,1.5]$。下面从右边压缩使得 $a_2=a_1$ 且 $b_2=c_1$。按这样的方法，可得到序列 $\{c_k\}$，它收敛到 $r\approx 1.114157141$。表 2.1 给出一个计算样本。

表 2.1　用二分法求解 $x\sin(x)-1=0$

k	左端点，a_k	中点，c_k	右端点，b_k	函数值 $f(c_k)$
0	0	1.	2.	-0.158529
1	1.0	1.5	2.0	0.496242
2	1.00	1.25	1.50	0.186231
3	1.000	1.125	1.250	0.015051
4	1.0000	1.0625	1.1250	-0.071827
5	1.06250	1.09375	1.12500	-0.028362
6	1.093750	1.109375	1.125000	-0.006643
7	1.1093750	1.1171875	1.1250000	0.004208
8	1.10937500	1.11328125	1.11718750	-0.001216
⋮	⋮	⋮	⋮	

二分法的优点是式(10)提供了一个对计算结果精度的预先估计。在例 2.7 中起始区间宽度为 $b_0-a_0=2$。假设表 2.1 继续执行到 31 个迭代，则根据(10)，误差边界为 $|E_{31}|\leqslant(2-0)/2^{32}\approx 4.656613\times10^{-10}$。因此 c_{31} 是 r 的近似值，精度为小数点后 9 位。重复二分法中的数 N 需要保证第 N 个中点 c_N 是零点的近似值，且误差不小于预定值 δ：

$$N=\operatorname{int}\left(\frac{\ln(b-a)-\ln(\delta)}{\ln(2)}\right) \tag{15}$$

该式的证明作为练习留给读者。

另一个常用的算法是试位法(method of false position)或写为 regula falsi method。由于二分法收敛速度相对较慢，因此试位法对它进行了改进。与上述条件一样，假设 $f(a)$ 和 $f(b)$ 符号相反。二分法使用区间 $[a,b]$ 的中点进行下一次迭代。如果找到经过点 $(a,f(a))$ 和 $(b,f(b))$ 的割线 L 与 x 轴的交点 $(c,0)$（如图 2.8 所示），则可得到一个更好的近似值。要寻找值 c，需定义线 L 的斜率 m 的两种表示，一种表示为：

$$m=\frac{f(b)-f(a)}{b-a} \tag{16}$$

这里使用了点 $(a,f(a))$ 和 $(b,f(b))$。另一种表示为：

$$m=\frac{0-f(b)}{c-b} \tag{17}$$

这里使用了点 $(c,0)$ 和 $(b,f(b))$。

使式(16)和式(17)的斜率相等，则有：

$$\frac{f(a)-f(a)}{b-a}=\frac{0-f(b)}{c-b}$$

为了更容易求解 c,可进一步表示为:

$$c=b-\frac{f(b)(b-a)}{f(b)-f(a)} \tag{18}$$

会出现三种与前面类似的可能性:

如果 $f(a)$和 $f(c)$的符号相反,则在$[a,c]$内有一个零点。 (19)

如果 $f(c)$和 $f(b)$的符号相反,则在$[c,b]$内有一个零点。 (20)

如果 $f(c)=0$,则 c 是零点。 (21)

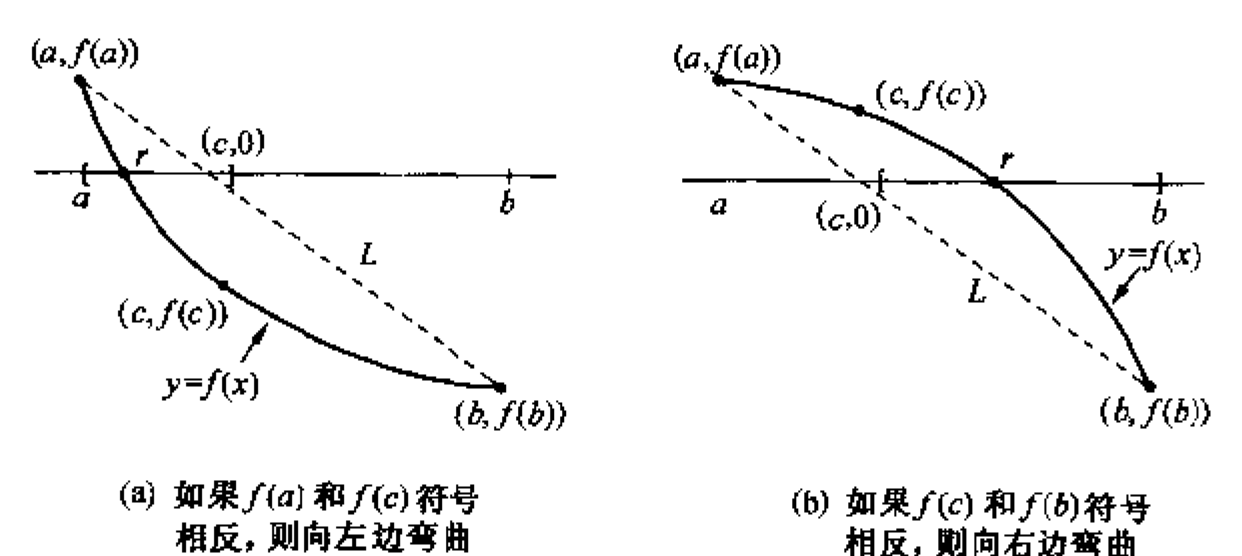


图 2.8 试值法的判定过程

2.2.2 试值法的收敛性

结合式(18),用式(19)和式(20)表示的判定过程可构造$\{a_n,b_n\}$区间序列,其中的每个序列包含零点。在每一步中,零点 r 的近似值为:

$$c_n=b_n-\frac{f(b_n)(b_n-a_n)}{f(b_n)-f(a_n)} \tag{22}$$

而且可以证明序列$\{c_n\}$将收敛到 r。但要注意,尽管区间宽度 b_n-a_n 越来越小,但它可能不趋近于0。例如曲线函数 $y=f(x)$在靠近点$(r,0)$处是凹形,一个端点是固定的,另一个点逼近解,但区间不趋近于零(如图 2.9 所示)。

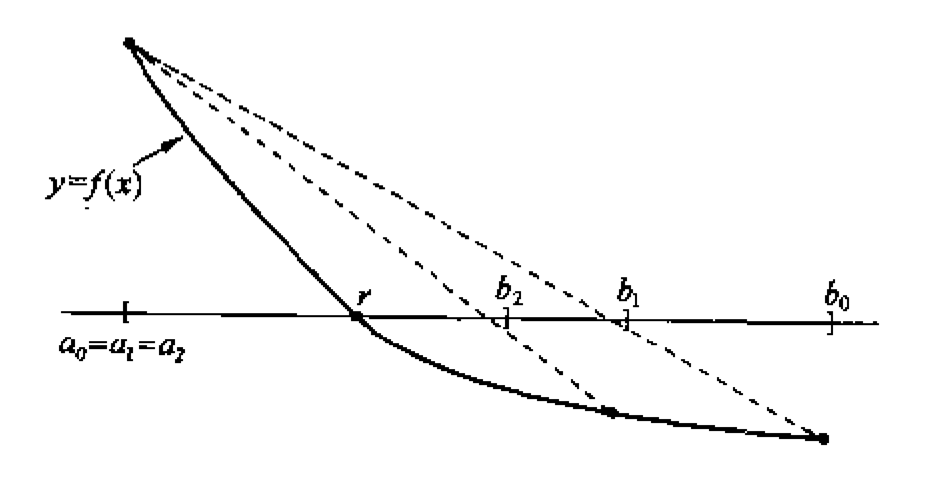


图 2.9 在试值法中的固定点

现在用试值法求解 $x\sin(x)-1=0$ 并观察它是否比二分法收敛得快。同时也要注意$\{b_n-a_n\}_{n=0}^{\infty}$不趋近于 0。

例 2.8 利用试值法寻找 $x\sin(x)-1=0$ 在区间 $[0,2]$ 内的根(函数 $\sin(x)$ 用弧度计算)。

初始值 $a_0=0$ 和 $b_0=2$。可得到 $f(0)=-1.00000000$ 和 $f(2)=0.81859485$,因此在区间 $[0,2]$ 间有一个根。利用式(22),可得到:

$$c_0=2-\frac{0.81859485(2-0)}{0.81859485-(-1)}=1.09975017 \text{ 和 } f(c_0)=-0.02001921$$

函数在区间 $[c_0,b_0]=[1.09975017,1]$ 改变符号,因此从左边压缩,设 $a_1=c_0=c_0$ 且 $b_1=b_0$。根据式(22)可得到下一个近似值:

$$c_1\approx 2-\frac{0.81859485(2-1.09975017)}{0.81859485-(-0.02001921)}=1.12124074$$

和

$$f(c_1)=0.00983461$$

接下来,$f(x)$ 在区间 $[a_1,c_1]=[1.09975017,1.12124074]$ 内改变符号,下一个判定是从右边压缩,且设 $a_2=a_1$ 和 $b_2=c_1$。整个计算过程如表 2.2 所示。

二分法的终止判别条件不适用于试值法,否则可能导致无穷循环。连续迭代的封闭性和 $|f(c_n)|$ 的值可同时用来作为程序 2.3 的终止判别条件。在 2.3 节将讨论这样做的原因。

表 2.2 用试值法求解 $x\sin(x)-1=0$

k	左端点,a_k	中点,c_k	右端点,b_k	函数值 $f(c_k)$
0	0.00000000	1.09975017	2.00000000	-0.02001921
1	1.09975017	1.12124074	2.00000000	0.00983461
2	1.09975017	1.11416120	1.12124074	0.00000563
3	1.09975017	1.11415714	1.11416120	0.00000000

程序 2.2(二分法) 求解方程 $f(x)=0$ 在区间 $[a,b]$ 内的一个根。前提条件是 $f(x)$ 是连续的,且 $f(a)$ 与 $f(b)$ 的符号相反

```
function [c,err,yc] = bisect(f,a,b,delta)
% Input  - f is the function input as a string 'f'
%        - a and b are the left and right end points
%        - delta is the tolerance
% Output- c is the zero
%        - yc = f(c)
%        - err is the error estimate for c
ya = feval(f,a);
yb = feval(f,b);
if ya * yb > 0,break,end
max1 = 1 + round((log(b - a) - log(delta))/log(2));
for k = 1:max1
    c = (a + b)/2;
    yc = feval(f,c);
    if yc == 0
        a = c;
        b = c;
```

```
elseif yb*yc>0
        b=c;
        yb=yc;
    else
        a=c;
        ya=yc;
    end
    if b-a < delta,break,end
end
c=(a+b)/2;
err=abs(b-a);
yc=feval(f,c);
```

程序 2.3(试位法) 求解方程 $f(x)=0$ 在区间 $[a,b]$ 内的根。前提条件是 $f(x)$ 是连续的,且 $f(a)$ 与 $f(b)$ 的符号相反

```
function [c,err,yc]=regula(f,a,b,deltak,epsilon,max1)
% Input  - f is the function input as a string 'f'
%        - a and b are the left and right end points
%        - delta is the tolerance for the zero
%        - epsilon is the tolerance for the value of f at the zero
%        - max1 is the maximum number of iterations
% Output- c is the zero
%        - yc=f(c)
%        - err is the error estimate for c
ya=feval(f,a);
yb=feval(f,b);
if ya*yb>0
    disp('Note:f(a)*f(b)>0'),
    break,
end
for k=1:max1
    dx=yb*(b-a)/(yb-ya);
    c=b-dx;
    ac=c-a;
    yc=feval(f,c);
    if yc==0,break;
    elseif yb*yc>0
        b=c;
        yb=yc;
    else
        a=c;
        ya=yc;
    end
    dx=min(abs(dx),ac);
    if abs(dx)<delta,break,end
    if abs(yc)<epsilon,break,end
end
c;
err=abs(b-a))/2;
yc=feval(f,c);
```

2.2.3 划分方法练习

在练习 1 和练习 2 中，如果在 240 个月内每月付款 P，求解满足整个年金 A 的利率 I。采用二分法和 I 的两个初始值计算下列 I 的 3 个近似值。

1. $P=\$275, A=\$250\,000, I_0=0.11, I_1=0.12$
2. $P=\$325, A=\$400\,000, I_0=0.13, I_1=0.14$
3. 对下面每个函数寻找一个区间 $[a,b]$，使得 $f(a)$ 和 $f(b)$ 的符号相反。
   (a) $f(x)=e^x-2-x$
   (b) $f(x)=\cos(x)+1-x$
   (c) $f(x)=\ln(x)-5+x$
   (d) $f(x)=x^2-10x+23$

在练习 4 到练习 7 中，利用试位法在区间 $[a_0,b_0]$ 内计算 c_0,c_1,c_2,c_3。

4. $e^x-2-x=0, [a_0,b_0]=[-2.4,-1.6]$
5. $\cos(x)+1-x=0, [a_0,b_0]=[0.8,1.6]$
6. $\ln(x)-5+x=0, [a_0,b_0]=[3.2,4.0]$
7. $x^2-10x+23=0, [a_0,b_0]=[6.0,6.8]$
8. 用 $[a_0,b_0],[a_1,b_1],\cdots,[a_n,b_n]$ 表示二分法产生的区间。
   (a)试证明 $a_0\leqslant a_1\leqslant\cdots\leqslant a_n\leqslant\cdots$ 和 $\cdots\leqslant b_n\leqslant\cdots\leqslant b_1\leqslant b_0$。
   (b)试证明 $b_n-a_n=(b_0-a_0)/2^n$。
   (c)设每个区间的中点为 $c_n=(a_n+b_n)/2$，试证明：
   $$\lim_{n\to\infty}a_n=\lim_{n\to\infty}c_n=\lim_{n\to\infty}b_n$$
   提示：回顾微积分参考书中单调序列的收敛性。
9. 如果用二分法求解函数 $f(x)=1/(x-2)$ 的零点，而且
   (a)区间为 $[3,7]$　　(b)区间为 $[1,7]$
   时，情况如何？
10. 如果用二分法求解函数 $f(x)=\tan(x)$ 的零点，而
   (a)区间为 $[3,4]$　　(b)区间为 $[1,3]$
   时，情况如何？
11. 假设用二分法寻找函数 $f(x)$ 在区间 $[2,7]$ 内的零点。执行多少次后可以使近似值 c_N 的精度达到 5×10^{-9}？
12. 试证明试位法的式(22)在代数上等价于：
   $$c_n=\frac{a_nf(b_n)-b_nf(a_n)}{f(b_n)-f(a_n)}$$
13. 构造用来确定二分法需要的迭代次数式(15)。提示：用 $|b-a|/2^{n+1}<\delta$ 和对数计算。
14. 多项式 $f(x)=(x-1)^3(x-2)(x-3)$ 有 3 个零点：3 个重根 $x=1$ 和单重根 $x=2$、$x=3$。如果 a_0 和 b_0 是任意两个实数，满足 $a_0<1$ 和 $b_0>3$，则 $f(a_0)f(b_0)<0$。这样，在区间 $[a_0,b_0]$ 内，二分法将收敛到 3 个零点之一。如果选择 $a_0<1$ 和 $b_0>3$，而且对任

意 $n \geqslant 1$,有 $c_n = \frac{a_n + b_n}{2}$ 不等于 1、2 或 3,则二分法一定不会收敛到哪个零点?为什么?

15. 如果多项式 $f(x)$ 在区间 $[a_0, b_0]$ 内有奇数个实零点,每个零点有奇数重根,则 $f(a_0)f(b_0) < 0$,且利用二分法将收敛到其中一个零点。如果 $a_0 < 1$ 且 $b_0 > 3$,而且对任意 $n \geqslant 1$,有 $c_n = \frac{a_n + b_n}{2}$ 不等于 $f(x)$ 的任意一个零点,则二分法一定不会收敛到哪个零点?为什么?

2.2.4 算法和程序

1. 如果在 240 个月内每月付款 \$300,求解满足整个年金 A 为 \$5000 000 的利率 I 的近似值(精确到小数点后 10 位)。
2. 设圆球由一种白橡树构成,密度 $\rho = 0.710$,半径 $r = 15$。将它放入水中,球浸入水中部分的质量(精确到小数点后 8 位)是多少?
3. 修改程序 2.2 和程序 2.3,使得输出分别类似于表 2.1 和表 2.2 的矩阵(即矩阵的第一行应当为 $[0\ a_0\ c_0\ b_0\ f(c_0)]$)。
4. 使用为求解问题 3 编写的程序,求解函数 $x = \tan(x)$ 的 3 个最小正根的近似值。
5. 一个单位球体被平面切成两部分,其中一部分的体积为另一部分的 3 倍。确定从球中心到平面的距离(精确到小数点后 10 位)。

2.3 初始近似值和收敛判定准则

划分方法依赖于寻找满足 $f(a)$ 与 $f(b)$ 符号相反的区间 $[a, b]$。一旦找到区间,无论区间多大,通过迭代总会找到一个根,因此这些方法被称为全局收敛法。然而,如果 $f(x) = 0$ 在区间 $[a, b]$ 有多个根,则必须使用不同的初始区间来寻找每个根,要寻找这些小区间并不容易。

在 2.4 节,将会研究牛顿拉夫申(Newton - Raphson)法和割线法以求解 $f(x) = 0$。这两种方法要求给定一个接近根的近似值以保证收敛性。因此这些方法被称为局部收敛法,局部收敛的速度远大于全局收敛的速度。一些混合方法首先采用全局收敛法,当迭代逼近根后,再切换到局部收敛法。

如果根的计算过程属于一个非常庞大的工程,那么可以采用一个较为简便的办法,即首先将函数画出来。通过对图 $y = f(x)$ 进行观察,并根据它的形状(凹性、斜率、振荡性、局部极值和拐点等)可以做出重要的判断。更重要的是,如果图中对应的点存在,它们可以被分析并用来决定根的近似值位置。这些近似值可作为求根算法的初始值。

求解过程必须非常仔细,计算机软件包中有各种复杂的图形软件。假设利用计算机对在区间 $[a, b]$ 内的函数 $y = f(x)$ 进行绘图,通常应将区间划分为 $N + 1$ 个等距点:$a = x_0 < x_1 < \cdots < x_N = b$,并计算函数值 $y_k = f(x_k)$。然后,或者使用线段,或者利用"拟合曲线"在连续点 (x_{k-1}, y_{k-1}) 和 (x_k, y_k) 之间进行绘图,其中 $k = 1, 2, \ldots, N$。必须确保有足够的点,才能保证当函数变化很快时曲线部分不丢失根。如果 $f(x)$ 连续,且两个邻接连续点 (x_{k-1}, y_{k-1}) 和 (x_k, y_k) 位于 x 轴的两边,则根据中值定理,在区间 $[x_{k-1}, x_k]$ 内至少有一个根。但如果在区间 $[x_{k-1}, x_k]$ 内有一个或多个靠得很近的根,而且邻接两点 (x_{k-1}, y_{k-1}) 和 (x_k, y_k) 位于 x 轴的同一边,则计算机生成的函数 f 的图形不能指示出适合中值定理的位置,即计算机产生的图形并

不是函数 f 实际图形的真实表示。当然,函数根非常接近的情况并不常见;在这种情况下,图形包含根的区域没有跨过 x 轴,或者根在纵向渐近线附近。当利用任何数值求根算法时,需要考虑到函数的这些特性。

最后,在两个非常接近的根或一个双重根附近,计算机在(x_{k-1},y_{k-1})和(x_k,y_k)之间生成的曲线可能不跨过或接触 x 轴。如果$|f(x_k)|$小于预定义值 $\epsilon(i,e.,f(x_k)\approx 0)$,则 x_k 是暂时的根的近似值。但在图中 x_k 附近有许多值接近 0,这样 x_k 可能并不接近实际的根。因此,必须增加要求:斜率在(x_k,y_k)附近改变符号,也就是说,保证 $m_{k-1}=\frac{y_k-y_{k-1}}{x_k-x_{k-1}}$和 $m_k=\frac{y_{k+1}-y_k}{x_{k+1}-x_k}$的符号一定相反。由于 $x_k-x_{k-1}>0$ 且 $x_{k+1}-x_k>0$,因此没有必要使用差商,通过检验 y_k-y_{k-1}的差和 $y_{k+1}-y_k$ 差符号是否相反就足够了。在这种情况下,x_k 是根的近似值。然而,不能保证这个初始值将一定会产生一个收敛序列。如果在 $y=f(x)$的图形中有一个局部最小(或最大)值趋近于 0,则当 $f(x_k)\approx 0$ 时,尽管 x_k 并不趋近一个根,但仍将 x_k 作为根的近似值。

例 2.9　在区间$[-1.2,1.2]$内寻找方程 $x^3-x^2-x+1=0$ 的根的近似值位置。为了说明情况,选择 $N=8$,并参见表 2.3。

考虑三个横坐标 -1.05、-0.3 和 0.9。因为 $f(x)$在区间$[-1.2,-0.9]$内改变符号,所以值 -1.05 是一个根的近似值;事实上,$f(-1.05)=-0.210$。
尽管在横坐标 0.3 附近斜率改变符号,但由于 $f(-0.3)=1.183$;因此 -0.3 不在根附近。
最后,函数的斜率在横坐标 0.9 附近改变符号,而且 $f(0.9)=0.019$,因此 0.9 是一个根的近似值(如图 2.10 所示)。

表 2.3　寻找根的近似值位置

x_k	函数值		M 的差值		左 $f(x)$或 $f'(x)$中的符号改变
	y_{k-1}	y_k	y_k-y_{k-1}	$y_{k+1}-y_k$	
-1.2	-3.125	-0.968	2.157	1.329	
-0.9	-0.968	0.361	1.329	0.663	在$[x_{k-1},x_k]$中 f 改变符号
-0.6	0.361	1.024	0.663	0.159	
0.3	1.024	1.183	0.159	-0.183	接近 x_k 时f'改变符号
0.0	1.183	1.000	-0.183	-0.363	
0.3	1.000	0.637	-0.363	-0.381	
0.6	0.637	0.256	-0.381	-0.237	
0.9	0.256	0.019	-0.237	0.069	接近 x_k 时f'改变符号
1.2	0.019	0.088	0.069	0.537	

2.3.1 检测收敛性

使用图形只能观察根的近似位置,必须用算法计算出计算机可接受的真正解 p_n。通常使用迭代产生序列$\{p_k\}$来逼近根 p,但必须提前设定终止迭代的判别条件或策略,使得计算机求出一个较精确的近似值后,可以停止计算。由于目标是求解 $f(x)=0$,所以最终值 p_n 应当满足$|f(p_n)|<\epsilon$。

用户可提供$|f(p_n)|$的允许误差 ϵ,然后通过迭代过程产生点 $P_k=(p_k,f(p_k))$,直到点 P_n 位于直线$y=+\epsilon$ 和$y=-\epsilon$ 之间水平区域,如图 2.11(a)所示。当用户求解 $h(x)=L$ 时,如果利用求根的算法求解 $f(x)=h(x)-L$,则此判定条件非常有用。

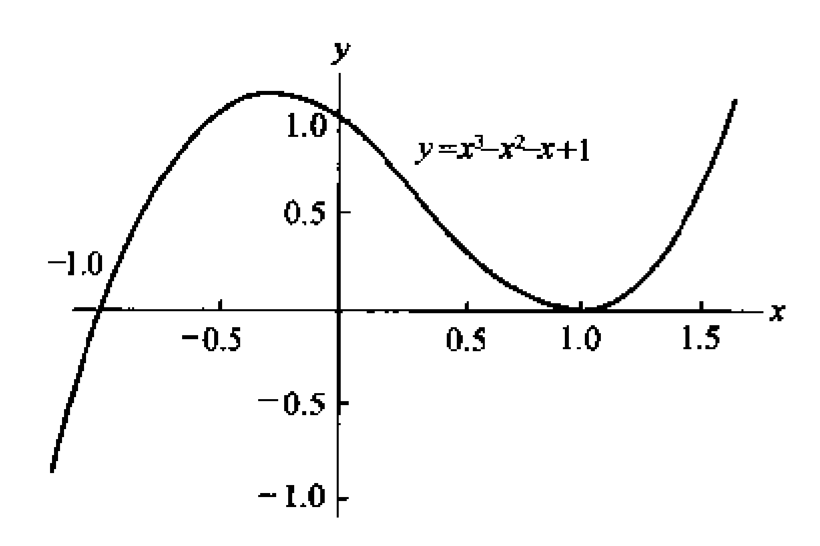


图 2.10 三次多项式 $y = x^3 - x^2 - x + 1$ 的图形

另一个终止判别条件与横坐标有关,可以用来判定序列 $\{p_k\}$ 是否收敛。如果在 $x = p$ 的两边画出垂直线 $x = p + \delta$ 和 $x = p - \delta$,当点 P_n 位于这两个垂直线之间时,可确定停止迭代,如图 2.11(b)所示。

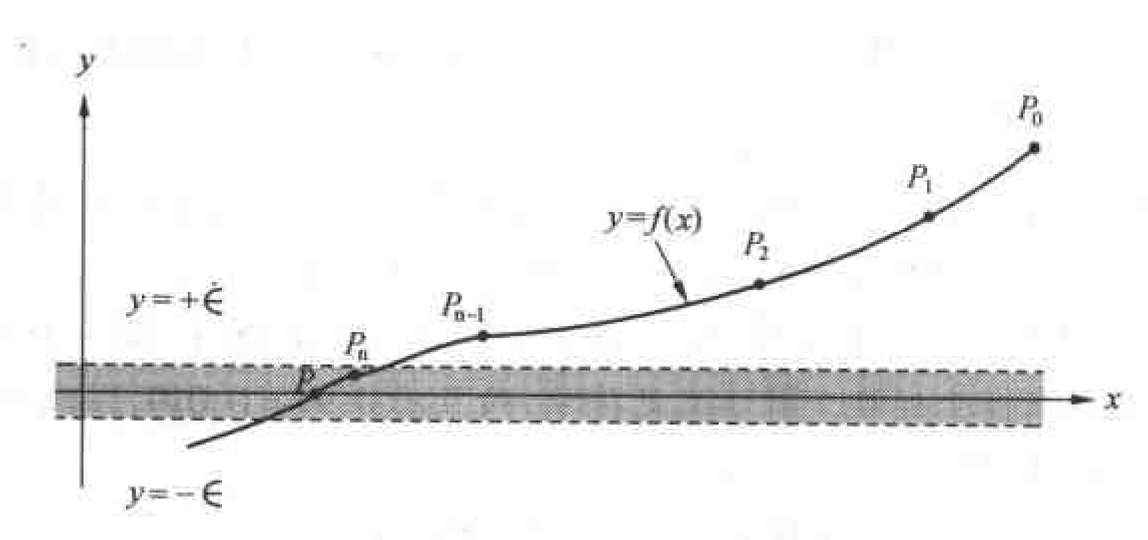


图 2.11(a) 定位函数 $f(x) = 0$ 的解的横向收敛区

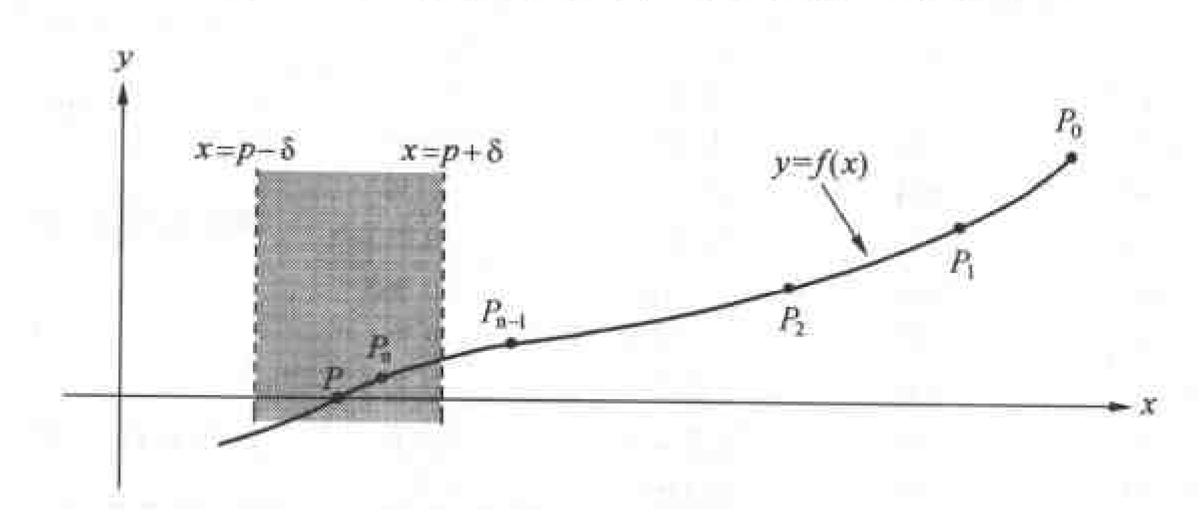


图 2.11(b) 定位函数 $f(x) = 0$ 的解的纵向收敛区

相比而言,后一个判别条件更能满足要求,但因为它包含不知道的解 p,所以实现起来较为困难。可以根据这个思路改进,即当连续迭代 p_{n-1} 和 p_n 足够接近,或者它们有 M 位有效数字时,停止进一步计算。

有些情况下,当 $p_n \approx p_{n-1}$ 或者 $f(p_n) \approx 0$ 时,就可以满足用户的算法。理解这一结论,需要借助于正确的逻辑推理。如果要求 $|p_n - p| < \delta$ 且 $|f(p_n) < \varepsilon|$,则点 P_n 位于包含根 $(p, 0)$ 的一个矩形区域内,如图 2.12(a)所示。如果规定 $|p_n - p| < \delta$ 或 $|f(p_n)| < \varepsilon$,则点 P_n 位于水平方向与垂直方向的并集区域内,如图 2.12(b)所示。允许误差 δ 和 ε 的大小很关键。如果允许误

差选得太小，则迭代可能无限执行下去。应当选择比 10^{-M} 大约大 100 倍的允许误差，这里 M 是计算机浮点数的小数位数。横坐标的封闭性可用如下判别条件检测：

$$|p_n - p_{n-1}| < \delta \qquad \text{（评价绝对误差）}$$

或

$$\frac{2|p_n - p_{n-1}|}{|p_n| - |p_{n-1}|} \qquad \text{（评价相对误差）}$$

纵坐标的封闭性通常通过 $|f(p_n) < \varepsilon|$ 来检查。

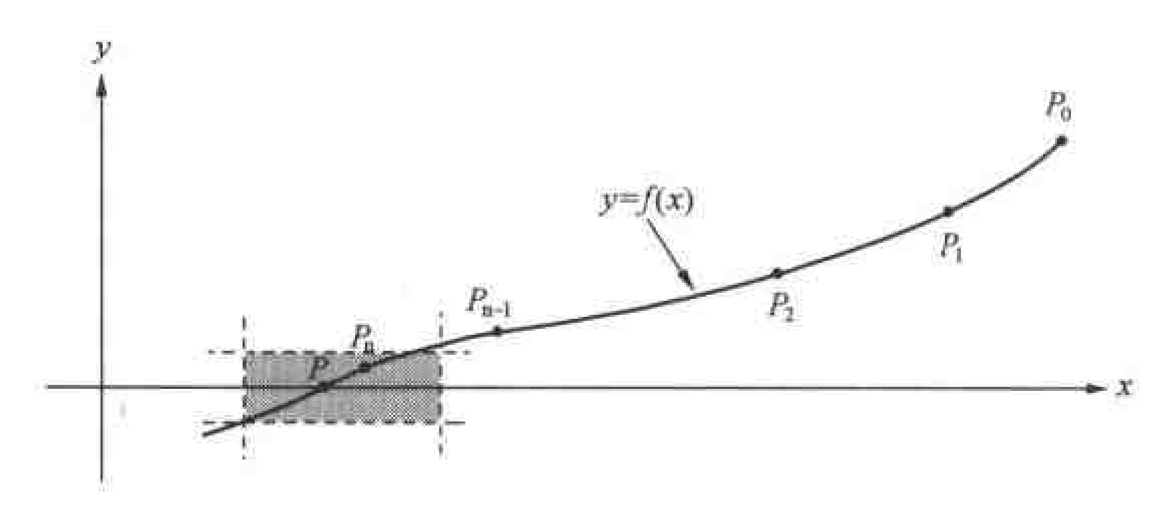


图 2.12(a)　由 $|x-p|<\delta$ 和 $|y|<\varepsilon$ 定义的矩形区域

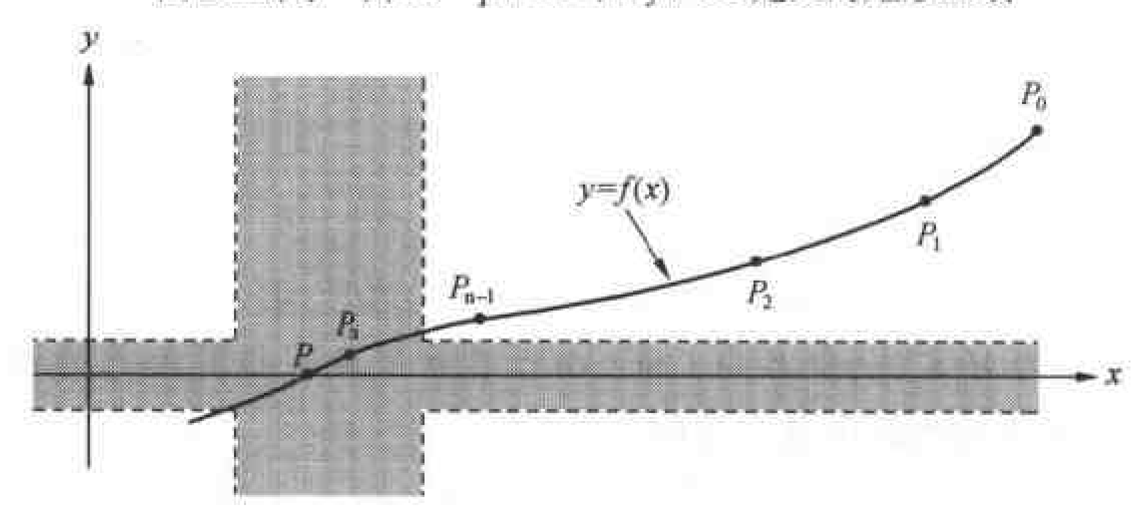


图 2.12(b)　由 $|x-p|<\delta$ 或 $|y|<\varepsilon$ 定义的无边界区域

2.3.2　有问题的函数(TroubleSome Functions)

由于截断误差和计算中的不稳定性，因此用计算机求解 $f(x)=0$ 总会有误差。如果图形 $y=f(x)$ 很陡地逼近根 $(p,0)$，则求根问题是良态的(well conditioned)，也就是说，很容易得到具有多位有效数字的解。如果图形 $y=f(x)$ 非常平缓地逼近根 $(p,0)$，则求根问题是病态的(ill conditioned)，也就是说，只能得到带少量有效数字的解。这种情况发生在 $f(x)$ 在 p 处有多个根的时候。这在下一节会进一步讨论。

程序 2.4(求解根近似值位置)　为了粗略估算方程 $f(x)=0$ 在区间 $[a,b]$ 内根的位置，使用等间隔采样点和如下的评定准则：

(i) $(y_{k-1})(y_k)<0$

(ii) $|y_k|<\varepsilon$ 且式 $(y_k-y_{k-1})(y_{k+1}-y_k)<0$

这样，要么 $f(x_{k-1})$ 与 $f(x_k)$ 符号相反，要么 $|f(x_k)|$ 足够小且曲线 $y=f(x)$ 的斜率在 $(x_k,f(x_k))$ 附近改变符号

```
function R = approot (X,epsilon)
% Input  - f is the object function saved as an M-file named f.m
%        - X is the vector of abscissas
%        - epsilon is the tolerance
% Output - R is the vector of approximate roots
Y = f(X);
yrange = max(Y) - min(Y);
epsilon2 = yrange * epsilon;
n = length(X);
m = 0;
X(n+1) = X(n);
Y(n+1) = Y(n);

for k = 2:n,
   if Y(k-1) * Y(k) < = 0,
      m = m + 1;
      R(m) = (X(k-1) + X(k))/2;
end
s = (Y(k) - Y(k-1)) *
    (Y(k+1) - Y(k));
if (abs(Y(k)) < epsilon2) & (s < = 0),
    m = m + 1;
    R(m) = X(k);
  end
end
```

例 2.10 使用程序 approot 求解函数 $f(x)=\sin(\cos(x^3))$ 在区间 $[-2,2]$ 内根的近似位置。首先将 f 保存为 M 文件,命名为 f.m。由于其结果被求根算法作为初始值,所以构造 X,使得近似值精确到小数点后面 4 位:

```
> > X = -2:.001:2;
> > approot(X,0.0001)
ans =
-1.9875  -1.6765  -1.1625 1.1625 1.6765 1.9875
```

通过将结果与函数 f 的图形进行比较,可以得到一个较好的初始近似值以用于求根算法。

2.3.3 初始近似值的练习

在练习题 1 到练习题 6 中,使用计算机或图形计算器,通过图形来确定函数 $f(x)=0$ 根的近似位置。在每个习题中,确定区间 $[a,b]$,以便利用程序 2.2 和程序 2.3 求解根(即 $f(a)f(b)<0$)。

1. $f(x)=x^2-e^x$ $\quad -2\leqslant x\leqslant 2$
2. $f(x)=x-\cos(x)$ $\quad -2\leqslant x\leqslant 2$
3. $f(x)=\sin(x)-2\cos(x)$ $\quad -2\leqslant x\leqslant 2$
4. $f(x)=\cos(x)+(1+x^2)^{-1}$ $\quad -2\leqslant x\leqslant 2$
5. $f(x)=(x-2)^2-\ln(x)$ $\quad 0.5\leqslant x\leqslant 4.5$
6. $f(x)=2x-\tan(x)$ $\quad -1.4\leqslant x\leqslant 1.4$

2.3.4 算法和程序

在问题1和问题2中，使用计算机或图形计算器，在给定的区间内，通过程序2.4求解实根的近似值，精确到小数点后4位。然后利用程序2.2和程序2.3，求解精确到小数点后包含12位的根的近似值。

1. $f(x)=1\,000\,000x^3-111\,000x^2+1110x-1$　　$-2\leqslant x\leqslant 2$
2. $f(x)=5x^{10}-38x^9+21x^8-5\pi x^6-3\pi x^5-5x^2+8x-3$　　$-15\leqslant x\leqslant 15$
3. 一个计算机程序使用点$(x_0,y_0),(x_1,y_1),\cdots$和$(x_N,y_N)$，可画出函数 $y=f(x)$的图形，通常还标记出图形的纵向高度，而且必须写出一个子程序来确定函数 f 在区间中的最大值和最小值。
   (a) 构造一个寻找值 $Y_{\max}=\max_k\{y_k\}$和 $Y_{\min}=\min_k\{y_k\}$的算法。
   (b) 写一个 MATLAB 程序寻找函数$f(x)$在区间$[a,b]$内根的近似位置和极值。
   (c) 使用(b)中的程序寻找问题1和问题2中根的位置和极值，并与真值进行比较。

2.4 牛顿拉夫申(Newton－Raphson)法和割线法

2.4.1 求根的斜率法

如果$f(x)$，$f'(x)$和$f''(x)$在根 p 附近连续，则可将它作为$f(x)$的特性，用于开发产生收敛到根 p 的序列$\{p_k\}$的算法。而且，这种算法产生序列$\{p_k\}$的速度比二分法和试位法快。牛顿拉夫申(简称牛顿)法依赖于$f'(x)$和$f''(x)$的连续性，是这类方法中已知的最有用和最好的方法之一。本小节首先通过图形方式对牛顿法进行介绍，然后用泰勒多项式对其进行更严格的分析。

设初始值 p_0 在根 p 附近。则函数 $y=f(x)$的图形与 x 轴相交于点$(p,0)$，而且点$(p_0,f(p_0))$位于靠近点$(p,0)$的曲线上(如图2.13所示)。将 p_1 定义为曲线在点$(p_0,f(p_0))$的切线与 x 轴的交点。则通过图2.13可以看到 p_1 比 p_0更靠近 p。如果写出如下所示的切线 L 的两种表达式，可得到与 p_1 和 p_0 相关的公式：

$$m=\frac{0-f(p_0)}{p_1-p_0}\tag{1}$$

上式是经过点$(p_1,0)$和点$(p_0,f(p_0))$的直线斜率，即：

$$m=f'(p_0)\tag{2}$$

上式是点$(p_0,f(p_0))$处的曲线斜率。式(1)和式(2)的斜率 m 相等，则求解 p_1 可得：

$$p_1=p_0-\frac{f(p_0)}{f'(p_0)}\tag{3}$$

重复上述过程可得，序列$\{p_k\}$收敛到 p。下面将精确定义上述计算过程。

定理2.5(牛顿拉夫申(Newton－Raphson)定理)　设$f\in C^2[a,b]$，且存在数 $p\in[a,b]$，满足$f(p)=0$。如果 $f'(p)\neq 0$，则存在一个数 $\delta>0$，使得由如下迭代定义序列$\{p_k\}_{k=0}^{\infty}$收敛到对于任意初始近似值 $p_0\in[p-\delta,p+\delta]$成立的 p：

$$p_k = g(p_{k-1}) = p_{k-1} - \frac{f(p_{k-1})}{f'(p_{k-1})}, \qquad k = 1,2,\cdots \tag{4}$$

注:函数 $g(x)$ 由如下公式定义:

$$g(x) = x - \frac{f(x)}{f'(x)} \tag{5}$$

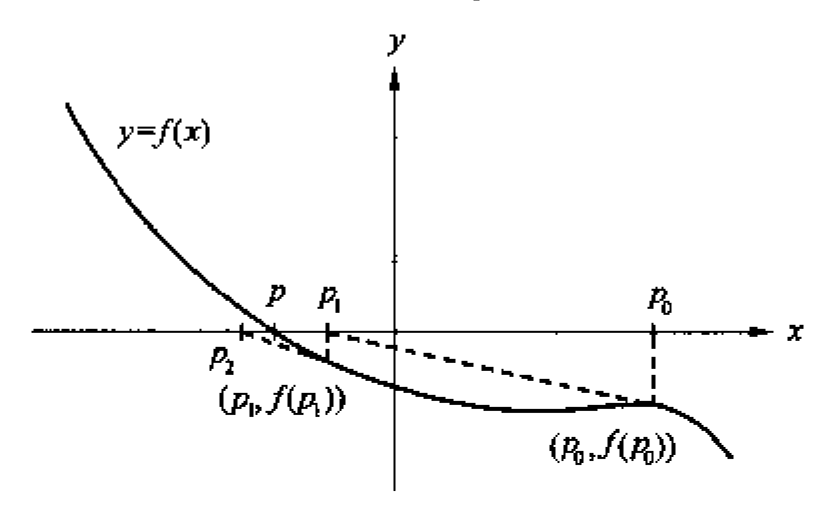


图 2.13 用于牛顿拉夫申法的 p_1 和 p_2 的几何结构

并称为牛顿拉夫申迭代函数。由于 $f(p)=0$,显然 $g(p)=p$。这样,通过寻找函数 $g(x)$ 的固定点,可以实现寻找方程 $f(x)=0$ 的根的牛顿拉夫申迭代。

证明:如图 2.13 所示的点 p_1 的几何结构不能帮助我们理解为何 p_0 需要靠近 p 或为何 $f''(x)$ 的连续性是必要的。这需要从一阶泰勒多项式和它的余项开始分析:

$$f(x) = f(p_0) + f'(p_0)(x-p_0) + \frac{f''(c)(x-p_0)^2}{2!} \tag{6}$$

这里,c 位于 p_0 和 x 之间。用 $x=p$ 代入式(6),并利用 $f(p)=0$,可得到:

$$0 = f(p_0) + f'(p_0)(p-p_0) + \frac{f''(c)(p-p_0)^2}{2!} \tag{7}$$

如果 p_0 足够逼近 p,则式(7)右边的最后一项比前两项的和小。因此最后一项可忽略,而且我们可利用如下近似表达式:

$$0 \approx f(p_0) + f'(p_0)(p-p_0) \tag{8}$$

求解式(8)中的 p,可得到 $p \approx p_0 - f(p_0)/f'(p_0)$。这可用来定义下一个根的近似值 p_1:

$$p_1 = p_0 - \frac{f(p_0)}{f'(p_0)} \tag{9}$$

当 p_{k-1} 用在式(9)的 p_0 位置上时,就可以建立一般规则(见式(4))。对大多数应用而言,这是需要理解的全部内容。但是,为了全面理解发生的情况,可能需要固定点迭代和应用定理 2.2。关键是对 $g'(x)$ 的分析:

$$g'(x) = 1 - \frac{f'(x)f'(x) - f(x)f''(x)}{(f'(x))^2} = \frac{f(x)f''(x)}{(f'(x))^2}$$

根据假设,$f(p)=0$,这样 $g'(p)=0$。由于 $g'(p)=0$,而且 $g'(x)$ 是连续的,所以可能找到一个数 $\delta>0$,在区间 $|g'(x)|<1$ 内满足定理 2.2 中的假设 $(p-\delta, p+\delta)$。因此,用 p_0 初始化收敛序列 $\{p_k\}_{k=0}^{\infty}$ 收敛到 $f(x)=0$ 的一个根的充分条件是 $p_0 \in (p-\delta, p+\delta)$,且 δ 满足:

$$\text{对所有的 } x \in (p-\delta, p+\delta) \text{ 有} \quad \frac{|f(x)f''(x)|}{|f'(x)|^2} < 1 \tag{10}$$

推论 2.2(求平方根的牛顿迭代) 设 A 为实数，且 $A>0$，而且令 $p_0>0$ 为 $\sqrt{A}$ 的初始近似值，可用下列递归规则定义序列 $\{p_k\}_{k=0}^{\infty}$。即：

$$p_k=\frac{p_{k-1}+\dfrac{A}{p_{k-1}}}{2},\qquad k=1,2,\cdots \tag{11}$$

则序列 $\{p_k\}_{k=0}^{\infty}$ 收敛到 $\sqrt{A}$，也可表示为 $\lim_{n\to\infty}p_k=\sqrt{A}$。

简单证明：从函数 $f(x)=x^2-A$，得到方程 $x^2-A=0$ 的根为 $\pm\sqrt{A}$。现在利用式(5)中的 $f(x)$ 和导数，可写出牛顿拉夫申迭代公式：

$$g(x)=x-\frac{f(x)}{f'(x)}=x-\frac{x^2-A}{2x} \tag{12}$$

此公式可简化为：

$$g(x)=\frac{x+\dfrac{A}{x}}{2} \tag{13}$$

用式(13)中的 $g(x)$ 定义式(4)中的递归迭代时，结果是式(11)。可以证明对任意初始值 $p_0>0$，式(11)中生成的序列将收敛。详细证明过程留作练习。

在推论 2.2 中，非常重要的一点是迭代函数 $g(x)$ 只包含算术符号 $+,-,\times$ 和/。如果 $g(x)$ 包含有平方根的计算，则会陷入循环推理中，即为了能够计算平方根，允许递归定义一个序列最终收敛到 $\sqrt{A}$。由于这个原因，选择了 $f(x)=x^2-A$，因为它只包含了算术操作。

例 2.11 用牛顿平方根算法求 $\sqrt{5}$ 的近似值。

从 $p_0=2$ 开始，计算：

$$p_1=\frac{2+5/2}{2}=2.25$$

$$p_2=\frac{2.25+5/2.25}{2}=2.236111111$$

$$p_3=\frac{2.236111111+5/2.236111111}{2}=2.236067978$$

$$P_4=\frac{2.36067978+5/2.236067978}{2}=2.236067978$$

进一步迭代可得到 $p_k\approx 2.236067978$，其中 $k>4$，收敛精度精确到小数点后面9位。

现在通过分析基础物理中的一个熟悉的问题，来分析为什么确定根的位置非常重要。设一个投射体从原点发射，仰角为 b_0，初始速度为 v_0。忽略空气阻力，如果用英尺测量，则飞行高度 $y=y(t)$ 和飞行水平行程 $x=x(t)$ 符合如下规则：

$$y=v_yt-16t^2 \text{ 和 } x=v_xt \tag{14}$$

这里，初始速度的水平分量为 $v_x=v_0\cos(b_0)$，垂直分量为 $v_y=v_0\sin(b_0)$。式(14)表示的数学模型容易用于求解投射体的飞行路径，但得出的飞行高度和飞行距离均高于实际。如果考虑到空气阻力与速度成一定比例，则运动方程变为：

$$y=f(t)=(Cv_y+32C^2)(1-e^{-t/C})-32Ct \tag{15}$$

和：

$$x = r(t) = Cv_x(1 - e^{-t/C}) \tag{16}$$

其中 $C = m/k$，k 是空气阻力的系数，m 是投射体的质量。如果 C 的值增大，则可得到更高的最高飞行高度和更远的飞行路程。考虑空气阻力的投射体飞行轨迹如图 2.14 所示。改进的模型更符合实际，但需要用求根算法求解 $f(t) = 0$ 来确定当投射体击中地面时经过的时间。式(14)中的基本模型不需要复杂的计算来求解飞行时间。

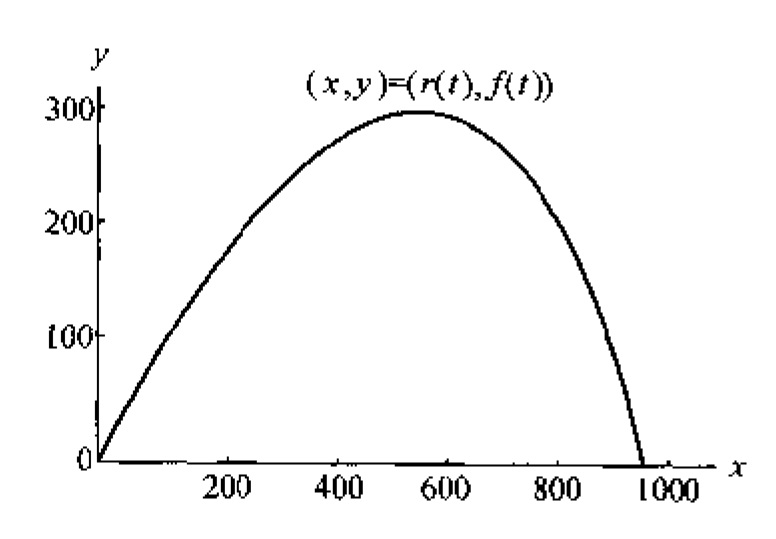


图 2.14 考虑空气阻力下的投射体飞行轨迹

例 2.12 一个投射体发射的仰角 $b_0 = 45°$，$v_y = v_x = 160$ ft/s，和 $C = 10$。求撞击地面后的飞行时间和飞行水平路程。

利用式(15)和式(16)，运动方程为 $y = f(t) = 4800(1 - e^{-t/10}) - 320t$ 和 $x = r(t) = 1600(1 - e^{-t/10})$。由于 $f(8) = 83.220972$ 和 $f(9) = -31.534367$，使用假定的初值 $p_0 = 8$，导数为 $f'(t) = 480e^{-t/10} - 320$，而且将 $f'(p_0) = f'(8) = -104.3220972$ 代入式(4)可得：

$$p_1 = 8 - \frac{83.22097200}{-104.3220972} = 8.797731010$$

计算情况如表 2.4 所示。

表 2.4 当高度 $f(t)$ 为 0 时求得的飞行时间

k	时间，p_k	$p_{k+1} - p_k$	高度，$f(p_k)$
0	8.00000000	0.79773101	83.22097200
1	8.79773101	−0.05530160	−6.68369700
2	8.74242941	−0.00025475	−0.03050700
3	8.74217467	−0.00000001	−0.00000100
4	8.74217466	0.00000000	0.00000000

值 p_4 的精度为小数点后面 8 位，飞行时间为 $t \approx 8.74217466$ 秒。飞行水平路程可用 $r(t)$ 计算，结果为：

$$r(8.74217466) = 1600(1 - e^{-0.874217466}) = 932.4986302 \text{ ft}$$

2.4.2 被零除错误

牛顿拉夫申法的一个明显缺陷是，如果在式(4)中 $f'(p_{k-1}) = 0$，则可能存在被零除错误。需有一个函数用于检查程序 2.5 中的这种情况，但在这种情况下，对于最后计算的近似值 p_{k-1}

如何处理？很可能 $f(p_{k-1})$ 足够接近零，这样 p_{k-1} 是根的一个可接受的近似值。下面将研究这种情况，并将发现一个有趣的事实，即迭代收敛的速度有多快。

定义 2.4(根的阶)　设 $f(x)$ 和它的导数在包含 $x=p$ 的某区间内有定义且连续。则称 $f(x)=0$ 在 $x=p$ 处根的阶为 M，当且仅当：

$$f(p)=0, f'(p)=0,\cdots,f^{(M-1)}(p)=0,\qquad 而且 f^{(M)}(p)\neq 0 \tag{17}$$

如果一个根的阶 $M=1$，则称此根为单根；如果一个根的阶 $M>1$，则称此根为重根。如果一个根的阶 $M=2$，则称此根为二重根，以此类推。下面的结论将说明这些概念。

引理 2.1　如果方程 $f(x)=0$ 在 $x=p$ 处有根的阶 M，则存在连续函数 $h(x)$，使得 $f(x)$ 可表示为：

$$f(x)=(x-p)^M h(x),\qquad h(p)\neq 0 \tag{18}$$

例 2.13　函数 $f(x)=x^3-3x+2$ 在 $p=-2$ 处有单根，在 $p=1$ 处有二重根。根据导数 $f'(x)=3x^2-3$ 和 $f''(x)=6x$ 可对其进行验证。当 $p=-2$，可得到 $f(-2)=0$ 和 $f'(-2)=9$，因此定义 2.4 中的 $M=1$，所以 $p=-2$ 是单根。当 $p=1$，可得到 $f(1)=0$，$f'(1)=0$ 和 $f''(1)=6$，因此定义 2.4 中的 $M=2$，所以 $p=1$ 是二重根。也可注意到 $f(x)$ 可因式分解为 $f(x)=(x+2)(x-1)^2$。

2.4.3 收敛速度

一个显著的性质是：如果 p 是 $f(x)=0$ 的单根，则牛顿法收敛很快，而且每次迭代结果的小数点后的精确位数大致上翻倍；另一方面，如果 p 是重根，每个连续的近似值误差是前一个误差的一小部分。为了使上述性质描述得更精确，可以通过定义收敛阶，测量序列的收敛速度。

定义 2.5(收敛阶)　设序列 $\{p_n\}_{n=0}^{\infty}$ 收敛到 p，而且当 $E_n=p-p_n$ 时，设 $n\geq 0$。如果两个常量 $A\neq 0$ 和 $R>0$ 存在，而且：

$$\lim_{n\to\infty}\frac{|p-p_{n+1}|}{|p-p_n|^R}=\lim_{n\to\infty}\frac{|E_{n+1}|}{|E_n|^R}=A \tag{19}$$

则序列称为以收敛阶 R 收敛到 p。数 A 称为渐进误差常数。当 $R=1,2$ 的情况为特殊情况：

如果 $R=1$，则称序列 $\{p_n\}_{n=0}^{\infty}$ 的收敛性为线性收敛。　(20)

如果 $R=2$，则称序列 $\{p_n\}_{n=0}^{\infty}$ 的收敛性为二次收敛。　(21)

如果 R 很大，序列 $\{p_n\}$ 快速收敛到 p；也就是说，关系式(19)意味着对于一个较大的值 n 有近似值 $|E_{n+1}|\approx A|E_n|^R$。例如，设 $R=2$ 且 $|E_n|\approx 10^{-2}$，则 $|E_{n+1}|\approx A\times 10^{-4}$。

一些序列的收敛率不是整数，下面例子中割线法中的收敛阶是 $R=(1+\sqrt{5})/2\approx 1.618033989$。

例 2.14(单根的二次收敛)　从 $p_0=-2.4$ 开始，用牛顿拉夫申迭代求多项式 $f(x)=x^3-3x+2$ 的根 $p=-2$。计算 $\{p_k\}$ 的迭代公式是：

$$p_k=g(p_{k-1})=\frac{2p_{k-1}^3-2}{3p_{k-1}^2-3} \tag{22}$$

用 $R=2$ 并利用公式(19)检查二次收敛,可得到表 2.5 中的值。

表 2.5 在一单根处用牛顿法收敛 4 次

k	p_k	$p_{k+1}-p_k$	$E_k=p-p_k$	$\frac{\lvert E_{k+1}\rvert}{\lvert E_k\rvert^2}$
0	-2.400000000	0.323809524	0.400000000	0.476190475
1	-2.076190476	0.072594465	0.076190476	0.619469086
2	-2.003596011	0.003587422	0.003596011	0.664202613
3	-2.000008589	0.000008589	0.000008589	
4	-2.000000000	0.000000000	0.000000000	

通过更详细地分析例 2.14 中的收敛速度,可发现每个连续迭代的误差是前一个迭代误差的一小部分,即:

$$|p-p_{k+1}|\approx A|p-p_k|^2$$

其中 $A\approx 2/3$。为了检查上式,利用:

$$|p-p_3|=0.000008589 \text{ 和 } |p-p_2|^2=|0.003596011|^2=0.000012931$$

而且容易看到:

$$|p-p_3|=0.000008589\approx 0.000008621=\frac{2}{3}|p-p_2|^2$$

例2.15(在二重根处线性收敛) 从 $p_0=1.2$ 开始,用牛顿拉夫申迭代求多项式 $f(x)=x^3-3x+2$ 的二重根 $p=1$。

用式(20)检查线性收敛,可得到表 2.6 中的值。

可以发现,牛顿拉夫申法收敛到二重根,但收敛速度慢。$f(p_k)$ 的值趋近于 0 的速度比 $f'(p_k)$ 的值要快,因此,当 $p_k\neq p$ 时,式(4)中的商 $f(p_k)/f'(p_k)$ 有定义。序列线性收敛,而且在每次迭代后,误差以 1/2 的比例下降。接下来的定理总结了牛顿法在单根和二重根上的性能。

表 2.6 在二重根处牛顿法线性收敛

k	p_k	$p_{k+1}-p_k$	$E_k=p-p_k$	$\frac{\lvert E_{k+1}\rvert}{\lvert E_k\rvert}$
0	1.200000000	-0.096969697	-0.200000000	0.515151515
1	0.103030303	-0.050673883	-1.103030303	0.508165253
2	1.052356420	-0.025955609	-0.052356420	0.496751115
3	1.026400811	-0.013143081	-0.026400811	0.509753688
4	1.013257730	-0.006614311	-0.013257730	0.501097775
5	1.006643419	-0.003318055	-0.006643419	0.500550093
⋮	⋮	⋮	⋮	⋮

定理 2.6(牛顿拉夫申迭代的收敛速度) 设牛顿拉夫申迭代产生的序列 $\{p_n\}_{n=0}^{\infty}$,收敛到函数 $f(x)$ 的根 p。如果 p 是单根,则是二次收敛,而且:

$$|E_{n+1}|\approx\frac{|f''(p)|}{2|f'(p)|}|E_n|^2,\qquad \text{其中 } n \text{ 足够大} \tag{23}$$

如果 p 是 M 阶多重根,则是线性收敛,而且:

$$|E_{n+1}|\approx\frac{M-1}{M}|E_n|,\qquad \text{其中 } n \text{ 足够大} \tag{24}$$

2.4.4 缺陷

被零除的错误很容易预见到，但另一种错误则不是那么容易被发现。设有函数 $f(x)=x^2-4x+5$，则由式(4)生成的实数序列 $\{p_k\}$ 将从左向右来回移动，不会收敛。通过简单分析可发现 $f(x)>0$，且无实根。

有时初始近似值 p_0 离要求的根太远，使得序列 $\{p_k\}$ 收敛到其他根上。当斜率 $f'(p_0)$ 很小，而且曲线 $y=f(x)$ 的切线接近垂直时，通常会发生这种情况。例如，如果 $f(x)=\cos(x)$ 而且求根 $p=\pi/2$，从 $p_0=3$ 开始，计算显示 $p_1=-4.01525255, p_2=-4.85265757, \cdots$，而且 $\{p_k\}$ 将收敛到另一个根 $-3\pi/2\approx-4.71238898$。

设 $f(x)$ 为正，在无限区间 $[a,\infty]$ 内单调递减，而且 $p_0>a$，则序列 $\{p_k\}$ 可能发散到 $+\infty$。例如，如果 $f(x)=xe^{-x}$，而且 $p_0=2.0$，则：

$$p_1=4.0,\ p_2=5.333333333,\ \cdots,\ p_{15}=19.723549434,\ \cdots$$

而且 $\{p_k\}$ 缓慢发散到 $+\infty$（如图 2.15(a)所示）。这个特殊的函数还有另一个令人惊讶的问题。当 x 变大时，$f(x)$ 的值迅速趋近于零，例如，$f(p_{15})=0.0000000536$，有可能错误地将 p_{15} 作为根。由于这个原因，需要设计程序 2.5 中的中止评定条件，其中包含相对误差 $2|p_{k+1}-p_k|/(|p_k|+10^{-6})$。当 $k=15$ 时，相对误差是 0.106817，因此允许误差 $\delta=10^{-6}$ 有助于保证不会产生一个错误的根。

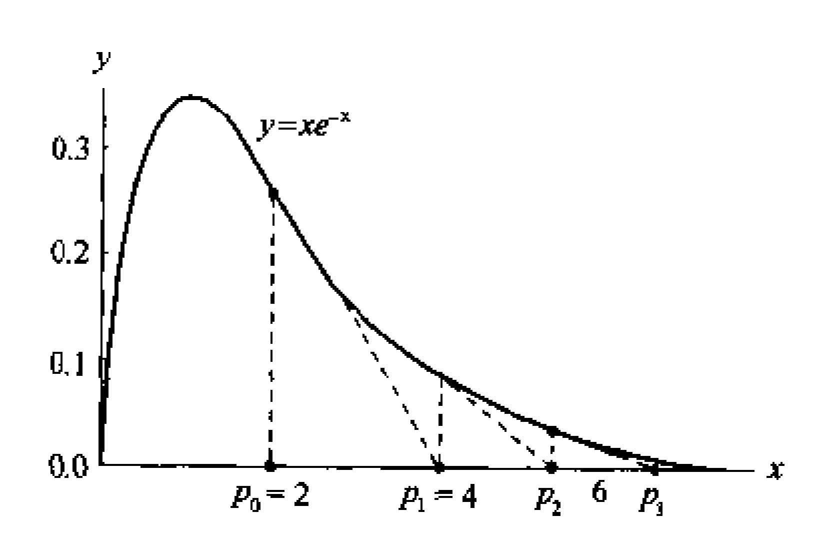


图 2.15(a)　求解 $f(x)=xe^{-x}$ 的牛顿拉夫申迭代产生离散序列

另一个现象是循环(cycling)，当序列 $\{p_k\}$ 中的项趋于重复或基本重复时，会发生这种现象。例如，如果 $f(x)=x^3-x-3$，而且初始近似值 $p_0=0$，则序列为：

$p_1=-3.000000,\quad p_2=-1.961538,\quad p_3=-1.147176,\quad p_4=-0.006579$

$p_5=-3.000389,\quad p_6=-1.961818,\quad p_7=-1.147430,\quad \cdots$

这里，我们陷入一个循环中，即当 $p_{k+4}\approx p_k$ 时，有 $k=0,1,\cdots$（如图 2.15(b)所示）。但如果初始值 p_0 足够逼近根 $p\approx1.671699881$，则序列 $\{p_k\}$ 收敛。如果 $p_0=2$，则序列收敛为：$p_1=1.72727272, p_2=1.67369173, p_3=1.671702570$ 和 $p_4=1.671699881$。

当 $|g'(x)|\geqslant1$ 在一个包含根 p 的区间内时，有可能发生离散振荡。例如，设 $f(x)=\arctan(x)$，则牛顿拉夫申迭代函数是 $g(x)=x-(1+x^2)\arctan(x)$ 和 $g'(x)=-2x\arctan(x)$。如果初始值 $p_0=1.45$，则：

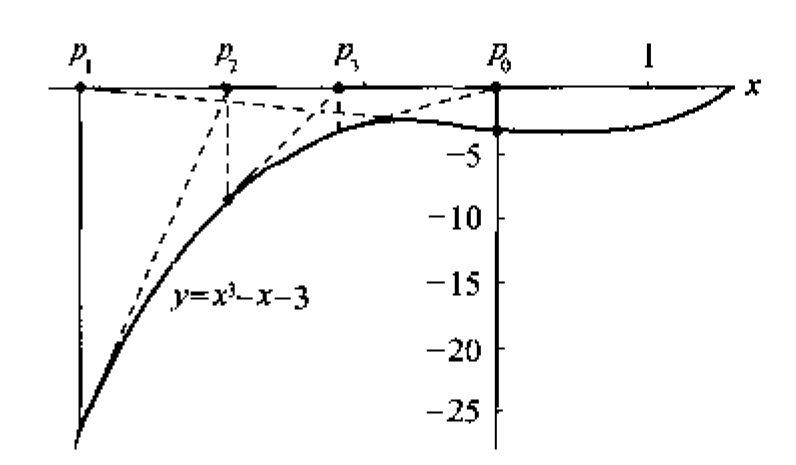


图 2.15(b) 求解函数 $f(x)=x^3-x-3$ 的牛顿拉夫申迭代产生一循环序列

$$p_1=-1.550263297, p_2=1.845931751, p_3=-2.889109054$$

依此类推(如图 2.15(c)所示)。如果初始值足够逼近根 $p=0$,则可得到一个收敛序列。如果 $p_0=0.5$,则:

$$p_1=-0.079559511, p_2=0.000335302, p_3=0.000000000$$

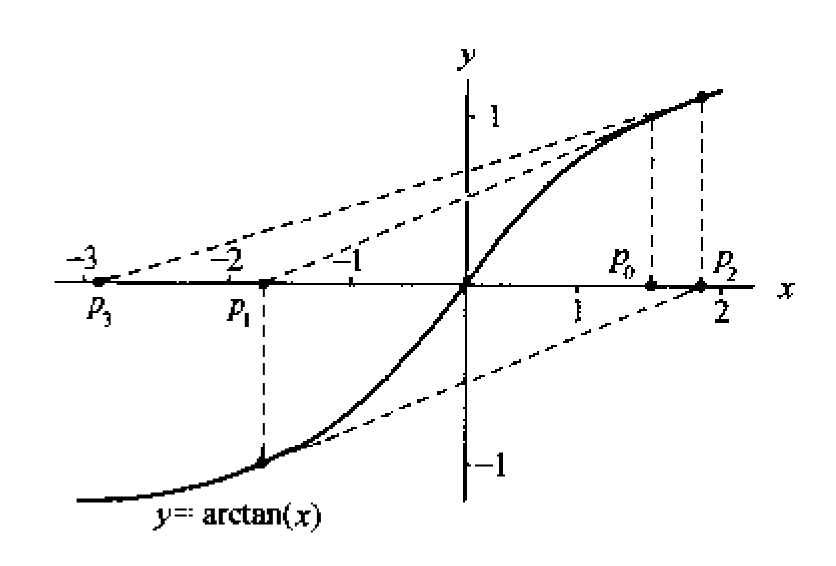


图 2.15(c) 求解函数 $f(x)=\arctan(x)$的牛顿拉夫申迭代产生一离散振荡序列

上述情况表明,必须按照真实情况说明结果:有时序列不会收敛,因为并不可能在 N 次迭代后总能找到结果。当求根算法没有找到根时需要发出警告。如果能借助于其他与问题相关的信息,则找到错误根的可能性会减少。有时 $f(x)$的根只在一个明确的区间内有意义。如果可以了解函数的行为或一个"精确"的图形,则可以更加容易地选择 p_0。

2.4.5 割线法

在牛顿拉夫申算法中每个迭代需要计算两个函数,$f(p_{k-1})$和 $f'(p_{k-1})$。以前计算基本函数的导数非常费功夫,但在现代计算机代数软件包的帮助下,这已不成问题。还有许多函数具有非基本项(累积、求和等),因此需要一种方法,它与牛顿法的收敛速度一样快,而且只计算 $f(x)$,不计算 $f'(x)$。割线法每步只计算一次 $f(x)$,而且在单根上的收敛阶 $R\approx1.618033989$。割线法与牛顿法差不多一样快,牛顿法的收敛阶为 2。

割线法包含的公式与试值法的公式一样,只是在关于如何定义每个后续项的逻辑判定上不一样。需要两个靠近点$(p,0)$的初始点$(p_0,f(p_0))$和$(p_1,f(p_1))$,如图 2.16 所示。将 p_2 定义为经过两个初始点的直线与 x 轴交点的横坐标,则图 2.16 显示出,p_2 比 p_0 或 p_1 更接近 p。

与 p_2,p_1 和 p_0 相关的表示斜率的方程如下:

$$m=\frac{f(p_1)-f(p_0)}{p_1-p_0} \text{和} m=\frac{0-f(p_1)}{p_2-p_1} \tag{25}$$

式(25)中的 m 值分别是经过两个初始点的割线的斜率和经过$(p_1,f(p_1))$与$(p_2,0)$的直线的斜率。设式(25)中两式的右边相等,并求解 $p_2=g(p_1,p_0)$,可得:

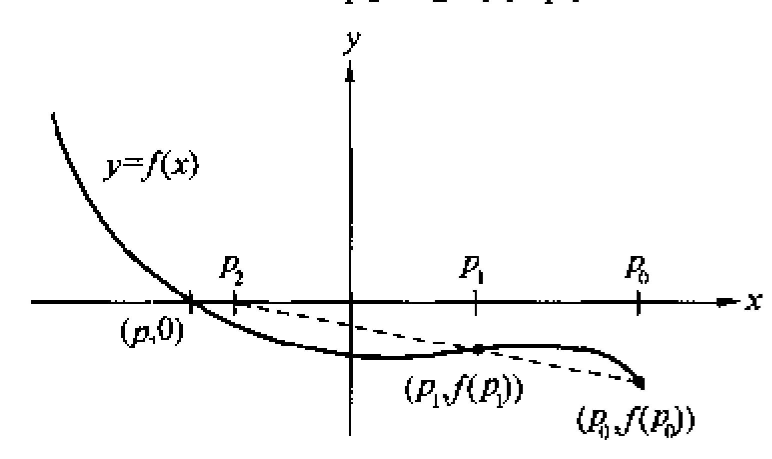


图 2.16 利用割线法时 p_2 的几何结构

$$p_2=g(p_1,p_0)=p_1-\frac{f(p_1)(p_1-p_0)}{f(p_1)-f(p_0)} \tag{26}$$

根据两点迭代公式可得到一般项为:

$$p_{k+1}=g(p_k,p_{k-1})=p_k-\frac{f(p_k)(p_k-p_{k-1})}{f(p_k)-f(p_{k-1})} \tag{27}$$

例 2.16(单根上的割线法) 从 $p_0=-2.6$ 和 $p_1=-2.4$ 开始,利用割线法求多项式函数 $f(x)=x^3-3x+2$ 的根 $p=-2$。

在此例中,迭代公式(27)是:

$$p_{k+1}=g(p_k,p_{k-1})=p_k-\frac{(p_k^3-3p_k+2)(p_k-p_{k-1})}{p_k^3-p_{k-1}^3-3p_k+3p_{k-1}} \tag{28}$$

可进一步化简为:

$$p_{k+1}=g(p_k,p_{k-1})=\frac{p_k^2p_{k-1}+p_kp_{k-1}^2-2}{p_k^2+p_kp_{k-1}+p_{k-1}^2-3} \tag{29}$$

迭代序列如表 2.7 所示。

表 2.7 在单根上割线法的收敛性

k	p_k	$p_{k+1}-p_k$	$E_k=p-p_k$	$\frac{\vert E_{k+1}\vert}{\vert E_k\vert^{1.618}}$
0	-2.600000000	0.200000000	0.600000000	0.914152831
1	-2.400000000	0.293401015	0.400000000	0.469497765
2	-2.106598985	0.083957573	0.106598985	0.847290012
3	-2.022641412	0.021130314	0.022641412	0.693608922
4	-2.001511098	0.001488561	0.001511098	0.825841116
5	-2.000022537	0.000022515	0.000022537	0.727100987
6	-2.000000022	0.000000022	0.000000022	
7	-2.000000000	0.000000000	0.000000000	

割线法和牛顿法之间有一定的关系。对于多项式函数 $f(x)$,如果用 $p_{k+1}=g(p_k,p_{k-1})$代替 p_k,则割线法的两点公式 p_{k-1}可归纳为牛顿法的一点公式 $p_{k+1}=g(p_k)$。事实上,如果在式

(29)中用 p_{k-1} 替代 p_k,则式(29)的右边与例 2.14 中式(22)的右边一致。

有关割线法收敛速度的证明可参见更详尽的数值积分教程。误差项满足关系式:

$$|E_{k+1}| \approx |E_k|^{1.618} \left|\frac{f''(p)}{2f'(p)}\right|^{0.618} \tag{30}$$

这里收敛阶 $R = (1+\sqrt{5})/2 \approx 1.618$,而且式(30)中的关系式只在单根情况下正确。

为了检验上述关系式,使用例 2.16 和特定值:

$$|p - p_5| = 0.000022537$$

$$|p - p_4|^{1.618} = 0.001511098^{1.618} = 0.000027296$$

和:

$$A = |f''(-2)/2f'(-2)|^{0.618} = (2/3)^{0.618} = 0.778351205$$

结合这些值,显然:

$$|p - p_5| = 0.000022537 \approx 0.000021246 = A|p - p_4|^{1.618}$$

2.4.6 加速收敛

当 p 是一个 M 阶根时,需要更好的求根技术以获得比线性收敛更快的速度。最终结果显示,通过对牛顿法进行改进,可使其在重根情况下的收敛阶为 2。

定理 2.7(牛顿拉夫申迭代的加速收敛) 设牛顿拉夫申算法产生的序列线性收敛到 M 阶根 $x = p$,其中 $M > 1$。则牛顿拉夫申迭代公式:

$$p_k = p_{k-1} - \frac{Mf(p_{k-1})}{f'(p_{k-1})} \tag{31}$$

将产生一收敛序列 $\{p_k\}_{k=0}^{\infty}$ 二次收敛到 p。

例 2.17(二重根情况下的加速收敛) 从 $p_0 = 1.2$ 开始,使用加速牛顿拉夫申迭代求函数 $f(x) = x^3 - 3x + 2$ 的二重根 $p = 1$。

由于 $M = 2$,加速公式(31)变成:

$$p_k = p_{k-1} - 2\frac{f(p_{k-1})}{f'(p_{k-1})} = \frac{p_{k-1}^3 + 3p_{k-1} - 4}{3p_{k-1}^2 - 3}$$

这样可得到表 2.8 中的值。

表 2.8 二重根情况下的加速收敛

k	p_k	$p_{k+1} - p_k$	$E_k = p - p_k$	$\frac{\lvert E_{k+1}\rvert}{\lvert E_k\rvert^2}$
0	1.200000000	−0.193939394	−0.200000000	0.151515150
1	1.006060606	−0.006054519	−0.006060606	0.165718578
2	1.000006087	−0.000006087	−0.000006087	
3	1.000000000	0.000000000	0.000000000	

表 2.9 对各种求根方法的收敛速度进行了比较,其中各个方法的常量 A 是不一样的。

表 2.9 收敛速度的比较

方法	特殊情况	连续误差的关系
二分法		$E_{k+1}\approx\frac{1}{2}\lvert E_k\rvert$
二分法		$E_{k+1}\approx A\lvert E_k\rvert$
割线法	重根	$E_{k+1}\approx A\lvert E_k\rvert$
牛顿拉夫申法	重根	$E_{k+1}\approx A\lvert E_k\rvert$
割线法	单根	$E_{k+1}\approx A\lvert E_k\rvert^{1.618}$
牛顿拉夫申法	单根	$E_{k+1}\approx A\lvert E_k\rvert^{2}$
加速牛顿拉夫申法	重根	$E_{k+1}\approx A\lvert E_k\rvert^{2}$

程序 2.5(牛顿拉夫申迭代) 使用初始近似值 p_0,利用迭代式 $p_k = p_{k-1} - \frac{f(p_{k-1})}{f'(p_{k-1})}$, $k=1,2,\cdots$,计算函数 $f(x)=0$ 的根的近似值

```
function [po,err,k,y] = newton(f,df,p0,delta,epsilon,max1)
% Input   - f is the object function input as a string'f'
%         - df is the derivative of f input as a string 'df'
%         - p0 is the initial approximation to a zero of f
%         - delta is the tolerance for p0
%         - epsilon is the tolerance for the function values y
%         - max1 is the maximum number of iterations
% Output  - p0 is the Newton - Raphson approximation to the zero
%         - err is the error estimate for p0
%         - k is the number of iterations
%         - y is the function value f(p0)
for k = 1:max1
   p1 = p0 - feval(f,p0)/feval(df,p0);
   err = abs(p1 - p0);
   relerr = 2 * err/(abs(p1) + delta);
   p0 = p1;
   y = feval(f,p0)
   if (err < delta)|(relerr < delta)|(abs(y) < epsilon),break,end
end
```

程序 2.6(割线法) 使用初始近似值 p_0 和 p_1,利用迭代式 $p_{k+1} = p_k - \frac{f(p_k)(p_k - p_{k-1})}{f(p_k) - f(p_{k-1})}$, $k=1,2,\cdots$,计算函数 $f(x)=0$ 的根的近似值

```
function [p1,err,k,y] = secant(f,p0,p1,delta,epsilon,max1)
% Input   - f is the object function input as a string'f'
%         - p0 and p1 are the initial approximations to a zero
%         - delta is the tolerance for p1
%         - epsilon is the tolerance for the function values y
%         - max1 is the maximum number of iterations
% Output  - p1 is the secant method approximation to the zero
```

```
%          - err is the error estimate for p1
%          - k is the number of iterations
%          - y is the function value f(p1)
for k = 1:max1
  p2 = p1 - feval(f,p1) * (p1 - p0)/(feval(f,p1) - feval(f,p0));
  err = abs(p2 - p1)
  relerr = 2 * err/(abs(p2) + delta);
  p0 = p1;
  p1 = p2;
  y = feval(f,p1);
  if(err < delta)|(relerr < delta)|(abs(y) < epsilon),break,end
end
```

2.4.7 牛顿拉夫申法和割线法的练习

对于某些需要计算的问题,可借助于计算器或计算机。

1. 设 $f(x)=x^2-x+2$
   (a) 求出牛顿拉夫申公式 $p_k=g(p_{k-1})$。
   (b) 从 $p_0=-1.5$ 开始,求 p_1,p_2 和 p_3。
2. 设 $f(x)=x^2-x-3$
   (a) 求出牛顿拉夫申公式 $p_k=g(p_{k-1})$。
   (b) 从 $p_0=1.6$ 开始,求 p_1,p_2 和 p_3。
   (c) 从 $p_0=0.0$ 开始,求 p_1,p_2,p_3 和 p_4。从这个序列可推测出什么?
3. 设 $f(x)=(x-2)^4$
   (a) 求出牛顿拉夫申公式 $p_k=g(p_{k-1})$。
   (b) 从 $p_0=2.1$ 开始,求 p_1,p_2,p_3 和 p_4。
   (c) 序列是二次收敛还是线性收敛?
4. 设 $f(x)=x^3-3x-2$
   (a) 求出牛顿拉夫申公式 $p_k=g(p_{k-1})$。
   (b) 从 $p_0=2.1$ 开始,求 p_1,p_2,p_3 和 p_4。
   (c) 序列是二次收敛还是线性收敛?
5. 设函数 $f(x)=\cos(x)$
   (a) 求出牛顿拉夫申公式 $p_k=g(p_{k-1})$。
   (b) 为了求根 $p=3\pi/2$,是否可采用 $p_0=3$? 为什么?
   (c) 为了求根 $p=3\pi/2$,是否可采用 $p_0=5$? 为什么?
6. 设函数 $f(x)=\arctan(x)$
   (a) 求出牛顿拉夫申公式 $p_k=g(p_{k-1})$。
   (b) 如果 $p_0=1.0$,则求 p_1,p_2,p_3 和 p_4。$\lim_{n\to\infty}p_k$ 是什么?
   (c) 如果 $p_0=2.0$,则求 p_1,p_2,p_3 和 p_4。$\lim_{n\to\infty}p_k$ 是什么?
7. 设函数 $f(x)=xe^{-x}$
   (a) 求出牛顿拉夫申公式 $p_k=g(p_{k-1})$。

(b) 如果 $p_0=0.2$,则求 p_1,p_2,p_3 和 p_4。$\lim_{n\to\infty}p_k$ 是什么?

(c) 如果 $p_0=20$,则求 p_1,p_2,p_3 和 p_4。$\lim_{n\to\infty}p_k$ 是什么?

(d) (c)中的 $f(p_4)$的值是多少?

在练习 8 到练习 10 中,利用割线法和式(27)计算接下来的两个迭代 p_2 和 p_3。

8. 设 $f^{(x)}=x^2-2x-1$,初始近似值 $p_0=2.6,p_1=2.5$。

9. 设 $f^{(x)}=x^2-x-3$,初始近似值 $p_0=1.7,p_1=1.67$。

10. 设 $f^{(x)}=x^3-x+2$,初始近似值 $p_0=-1.5,p_1=-1.52$。

11. 立方根算法(Cube - root algorithm)。函数为 $f(x)=x^3-A$。假定 A 是任意实数,推导递归公式:

$$p_k=\frac{2p_{k-1}+A/p_{k-1}^2}{3},\quad k=1,2,\cdots$$

12. 设 $f(x)=x^N-A$,这里 N 是正整数

(a) 对于不同的 N 和A,方程 $f(x)=0$ 的实数解是什么?

(b)推导寻找 A 的第N 个根的递归公式:

$$p_k=\frac{(N-1)p_{k-1}+A/P_{k-1}^{N-1}}{N},\quad k=1,2,\cdots$$

13. 如果 $f(x)=x^2-14x+50$,能否用牛顿拉夫申迭代求解 $f(x)=0$? 为什么?

14. 如果 $f(x)=x^{1/3}$,能否用牛顿拉夫申迭代求解 $f(x)=0$? 为什么?

15. 如果 $f(x)=(x-3)^{1/2}$,而且初始值 $p_0=4$,能否用牛顿拉夫申迭代求解 $f(x)=0$? 为什么?

16. 试推导出(11)中序列的极限。

17. 证明定理 2.5 的式(4)中的序列$\{p_k\}$收敛到 p,使用如下步骤。

(a) 证明如果 p 是式(5)中 $g(x)$的固定点,则 p 是$f(x)$的零点。

(b) 如果 p 是$f(x)$的零点而且 $f'(p)\neq 0$,证明 $g'(p)=0$。利用这个结论和定理 2.3,证明式(4)中的序列$\{p_k\}$收敛到 p。

18. 证明定理 2.6 中的式(23)。使用如下步骤。根据定理 1.11,在 $x=p_k$ 处展开$f(x)$得到:

$$f(x)=f(p_k)+f'(p_k)(x-p_k)+\frac{1}{2}f''(c_k)(x-p_k)^2$$

由于 p 是$f(x)$的零点,设 $x=p$ 可得到:

$$0=f(p_k)+f'(p_k)(p-p_k)+\frac{1}{2}f''(c_k)(p-p_k)^2$$

(a) 假设对于靠近根 p 的所有 x 有$f'(x)\neq 0$。利用上述事实和 $f'(p_k)\neq 0$ 证明:

$$p-p_k+\frac{f(p_k)}{f'(p_k)}=\frac{-f''(c_k)}{2f'(p_k)}(p-p_k)^2$$

(b) 设 $f'(x)$和 $f''(x)$变化的速度不快,所以可用近似值 $f'(p_k)\approx f'(p)$和 $f''(c_k)\approx f''(p)$。利用(a)可得到:

$$E_{k+1}\approx\frac{-f''(p)}{2f'(p)}E_k^2.$$

19. 设 A 为正实数。

(a) 证明 A 可表达为 $A = q \times 2^{2m}$,其中 $1/4 \leqslant q < 1$,且 m 为整数。

(b) 利用(a)证明平方根是 $A^{1/2} = q^{1/2} \times 2^m$。注:让 $p_0 = (2q+1)/3$,其中$1/4 \leqslant q < 1$,并利用牛顿法的式(11)。经过三个迭代,p_3 是 $q^{1/2}$的近似值,精度为二进制小数点后 24 位。这是计算机硬件计算平方根的常用算法。

20. (a) 证明割线法的式(27)在算术上等价于:

$$p_{k+1} = \frac{p_{k-1}f(p_k) - p_kf(p_{k-1})}{f(p_k) - f(p_{k-1})}$$

(b) 试解释为什么上式中减法导致精度丧失,使得此式在数值计算上不如式(27)。

21. 设 p 是函数$f(x) = 0$ 的根,p 的阶 $M = 2$。证明加速牛顿拉夫申迭代:

$$p_k = p_{k-1} - \frac{2f(p_{k-1})}{f'(p_{k-1})}$$

二次收敛(参见练习题 18)。

22. 哈雷法(Halley's method)是加速牛顿法收敛的另一个途径。哈雷(Halley)迭代公式是:

$$g(x) = x - \frac{f(x)}{f'(x)}\left(1 - \frac{f(x)f''(x)}{2(f'(x))^2}\right)^{-1}$$

括号中的项是对牛顿拉夫申公式的改进。哈雷法在$f(x)$的单根情况下可达到三次收敛($R = 3$)。

(a) 设函数 $f(x) = x^2 - A$,试求出哈雷迭代公式 $g(x)$,以便求解$\sqrt{A}$。用 $p_0 = 2$ 来近似$\sqrt{5}$,并计算 p_1, p_2 和 p_3。

(b) 设函数 $f(x) = x^3 - 3x + 2$,求哈雷迭代公式 $g(x)$。利用 $p_0 = -2.4$ 计算 p_1, p_2 和 p_3。

23. 用于重根情况的改进牛顿拉夫申法。如果 p 是 M 阶重根,则 $f(x) = (x-p)^M q(x)$,其中$q(p) \neq 0$。

(a) 证明 $h(x) = f(x)/f'(x)$在 p 处有单根。

(b) 证明当用牛顿拉夫申法求 $h(x)$的根时,$g(x) = x - h(x)/h'(x)$变成:

$$g(x) = x - \frac{f(x)f'(x)}{(f'(x))^2 - f(x)f''(x)}$$

(c) (b)中利用 $g(x)$的迭代二次收敛到 p,解释为何发生这种情况。

(d) 0 是函数 $f(x) = \sin(x^3)$的三重根。从 $p_0 = 1$ 开始,利用改进牛顿拉夫申法计算 p_1, p_2 和 p_3。

24. 设一求解 $f(x) = 0$ 的迭代方法产生如下四个连续误差项(参见例 2.11):$E_0 = 0.400000, E_1 = 0.043797, E_2 = 0.000062$ 和 $E_3 = 0.000000$。估算由迭代法生成的渐进误差常数 A 和序列的收敛阶 R。

2.4.8 算法和程序

1. 修改程序 2.5 和程序 2.6,使得程序在下列情况下能够适当地显示错误信息:

(i) 在式(4)或式(27)中分别发生被零除的情况。

(ii) 超过迭代次数 max1。

2. 显示由式(4)和式(27)生成的序列中的项,通常具有启发性(如表 2.4 中的第二列)。修改程序 2.5 和程序 2.6,使其能够分别显示由式(4)和式(27)生成的序列。
3. 利用牛顿平方根算法修改程序 2.5,并用其近似计算下列每个平方根到小数点后 10 位。

   (a) $p_0=3$,求$\sqrt{8}$的近似值。

   (b) $p_0=10$,求$\sqrt{91}$的近似值。

   (c) $p_0=-3$,求 $-\sqrt{8}$的近似值。
4. 用练习 11 中的立方根算法修改程序 2.5,并用其近似计算下列每个立方根到小数点后 10 位。

   (a) $p_0=2$,求 $7^{1/3}$的近似值。

   (b) $p_0=6$,求 $200^{1/3}$的近似值。

   (c) $p_0=-2$,求$(-7)^{1/3}$的近似值。
5. 利用定理 2.7 中的加速牛顿拉夫申算法修改程序 2.5,并用其求下列函数的 M 阶根 p 的近似值。

   (a) $f(x)=(x-2)^5, M=5, p=2$　　初始值 $p_0=1$。

   (b) $f(x)=\sin(x^3), M=3, p=0$　　初始值 $p_0=1$。

   (c) $f(x)=(x-1)\ln(x), M=2, p=1$　　初始值 $p_0=2$。
6. 利用练习 22 中的哈雷法修改程序 2.5,并用其求解函数 $f(x)=x^3-3x+2$ 的单根 $p_0=-2.4$。
7. 设投射体的运动方程为:
$$y=f(t)=9600(1-e^{-t/15})-480t$$
$$x=r(t)=2400(1-e^{-t/15})$$
   (a) 求当撞击地面时经过的时间,精确到小数点后 10 位。

   (b) 求水平飞行路程,精确到小数点后 10 位。
8. (a) 求最接近点(3,1)的抛物线 $y=x^2$ 上的点,精确到小数点后 10 位。

   (b) 求最接近点(2.1,0.5)的曲线函数 $y=\sin(x-\sin(x))$上的点,精确到小数点后 10 位。

   (c) 求曲线函数 $f(x)=x^2+2$ 与曲线函数 $g(x)=(x/5)-\sin(x)$之间的最小垂直距离处的 x 值,精确到小数点后 10 位。
9. 一个敞口盒由 10×16 英寸的长方形金属片构成。如果要求盒子的容积为 100 立方英寸,则需要从盒子的边角处砍去多大尺寸的正方形(精确到 0.000000001 英寸)。
10. 悬链线由悬挂的绳索构成。设最低点为(0,0),则悬链线的公式为 $y=C\cosh(x/C)-C$,为确定经过$(\pm a,b)$的悬链线,需要求解方程 $b=C\cosh(a/C)-C$ 得到 C。

    (a) 证明经过$(\pm 10,6)$的悬链线上 $y=9.1889\cosh(x/9.1889)-9.1889$。

    (b) 求解经过$(\pm 12,5)$的悬链线。

2.5 Aitken 过程、Steffensen 法和 Muller 法(可选)

在 2.4 节中,可看到在重根情况下,牛顿法收敛很慢,而且迭代序列$\{p_k\}$是线性收敛。定

理 2.7 显示了如何加速收敛,但需要预先知道根的阶。

2.5.1 Aitken 过程

一种称为 Aitken's Δ^2 过程的技术可加速任何线性收敛的序列。为此,有如下定义:

定义 2.6 设有序列$\{p_n\}_{n=0}^{\infty}$,用如下表达式定义前向微分 Δp_n:

$$\Delta p_n = p_{n+1} - p_n, n \geqslant 0 \tag{1}$$

高阶 $\Delta^k p_n$ 可递归定义为:

$$\Delta^k p_n = \Delta^{k-1}(\Delta p_n), k \geqslant 2 \tag{2}$$

定理 2.8(Aitken 加速) 设序列$\{p_n\}_{n=0}^{\infty}$线性收敛到极限 p,而且对所有 $n \geqslant 0$,有 $p - p_n \neq 0$。如果存在实数 A,且$|A| < 1$,满足:

$$\lim_{n \to \infty} \frac{p - p_{n+1}}{p - p_n} = A \tag{3}$$

则定义为:

$$q_n = p_n - \frac{(\Delta p_n)^2}{\Delta^2 p_n} = p_n - \frac{(p_{n+1} - p_n)^2}{p_{n+2} - 2p_{n+1} + p_n} \tag{4}$$

的序列$\{q_n\}_{n=0}^{\infty}$收敛到 p,且比$\{p_n\}_{n=0}^{\infty}$快,而且:

$$\lim_{n \to \infty} \left| \frac{p - q_n}{p - p_n} \right| = 0 \tag{5}$$

证明:下面将证明如果得到式(4),并把对式(5)的证明作为练习。由于式(3)中的项是逼近一个极限,可写成:

$$\frac{p - p_{n+1}}{p - p_n} \approx A \quad 和 \quad \frac{p - p_{n+2}}{p - p_{n+1}} \approx A, \quad 其中\ n\ 足够大 \tag{6}$$

则根据式(6)中的关系式可得到:

$$(p - p_{n+1})^2 \approx (p - p_{n+2})(p - p_n) \tag{7}$$

当展开式(7)的两边并消除 p^2,可得到:

$$p \approx \frac{p_n + 2p_n - p_{n+1}^2}{p_{n+2} - 2p_{n+1} + p_n} = q_n, \quad n = 0, 1, \cdots \tag{8}$$

用式(8)来定义项 q_n。可重新对其变换得到式(4),使得用计算机计算时可得到更小的误差传播。

例 2.18 证明例 2.2 中的序列$\{p_n\}$是线性收敛。同时证明由 Aitken's Δ^2 过程得到的序列$\{q_n\}$收敛得更快。

使用函数 $g(x) = e^{-x}$,从 $p_0 = 0.5$ 开始,通过固定点迭代可得到序列$\{p_n\}$。收敛后的极限为 p_n 和 q_n 的值如表 2.10 和表 2.11 所示。例如,q_1 的值的计算过程如下:

$$\begin{aligned} q_1 &= p_1 - \frac{(p_2 - p_1)^2}{p_3 - 2p_2 + p_1} \\ &= 0.606530660 - \frac{(-0.061291448)^2}{0.095755331} = 0.567298989 \end{aligned}$$

表 2.10　线性收敛序列 $\{p_n\}$

n	p_n	$E_n = p_n - p$	$A_n = \frac{E_n}{E_{n-1}}$
1	0.606530660	0.039387369	-0.586616609
2	0.545239212	-0.021904079	-0.556119357
3	0.579703095	0.012559805	-0.573400269
4	0.560064628	-0.007078663	-0.563596551
5	0.571172149	0.004028859	-0.569155345
6	0.564862947	-0.002280343	-0.566002341

表 2.11　用 Aitken 过程得到的序列 $\{q_n\}$

n	q_n	$q_n - p$
1	0.567298989	0.000155699
2	0.567193142	0.000049852
3	0.567159364	0.000016074
4	0.567148453	0.000005163
5	0.567144952	0.000001662
6	0.567143825	0.000000534

尽管表 2.11 中的序列 $\{q_n\}$ 为线性收敛，根据定理 2.8，它比 $\{p_n\}$ 收敛得快。而通常 Aitken 方法改进后收敛速度更快。把固定点迭代和 Aitken 过程结合起来的方法称为 Steffensen 加速，这在程序 2.7 和练习中有更详细的描述。

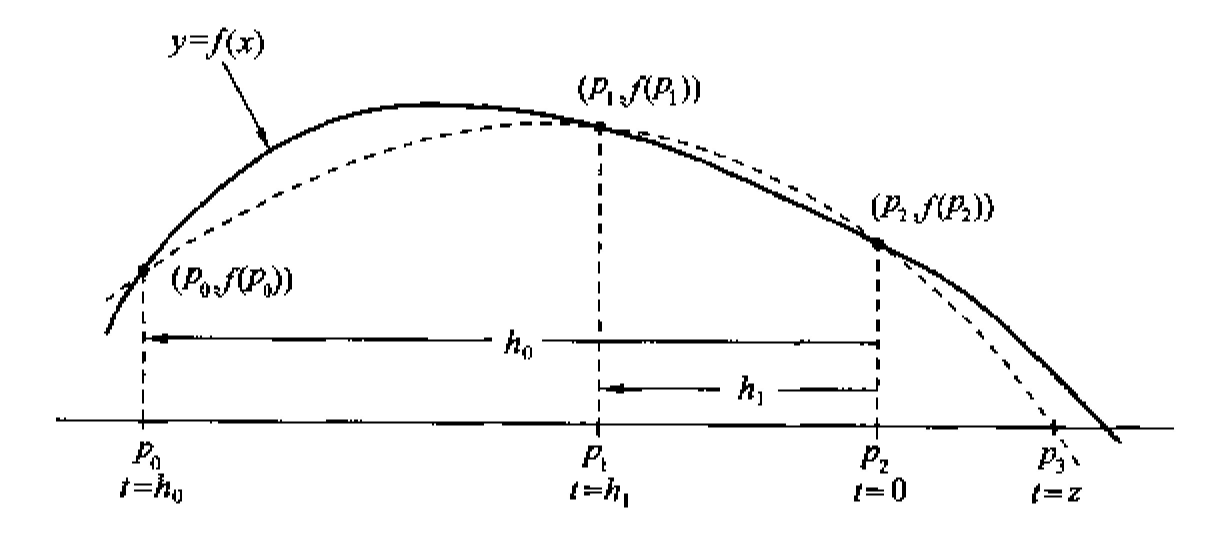


图 2.17　采用 Muller 法的初始近似值 p_0，p_1 和 p_2，以及差分 h_0 和 h_1

2.5.2 Muller 法

由于 Muller 法不需要计算函数的导数，可把 Muller 法看成割线法的广义形式。Muller 法是一种迭代方法，需要三个初始点 $(p_0, f(p_0))$，$(p_1, f(p_1))$ 和 $(p_2, f(p_2))$。构造一条抛物线经过这三点，然后利用二次函数求下一个近似值的二次根。可以证明当接近一个单根时，Muller 法比割线法收敛得快，而且基本上与牛顿法一样快。此方法可用来求函数的实数零点和复数零点，而且可用于为复杂的算式编制程序。

不失一般性，设 p_2 是根的最佳近似值，并设一个抛物线经过 3 个初始点，如图 2.17 所示。改变变量：

$$t = x - p_2 \tag{9}$$

使用差分为:

$$h_0 = p_0 - p_2 \text{ 和 } h_1 = p_1 - p_2 \tag{10}$$

设包含变量 t 的二次多项式为:

$$y = at^2 + bt + c \tag{11}$$

根据每一点可得到一个包含 a、b 和 c 的方程:

$$\begin{aligned} &\text{当 } t = h_0: && ah_0^2 + bh_0 + c = f_0 \\ &\text{当 } t = h_1: && ah_1^2 + bh_1 + c = f_1 \\ &\text{当 } t = 0: && ao^2 + b0 + c = f_2 \end{aligned} \tag{12}$$

从式(12)中的第三个方程,可看到:

$$c = f_2 \tag{13}$$

将式(13)代入式(12)中的前两个方程,并利用定义 $e_0 = f_0 - c$ 和 $e_1 = f_1 - c$,可得到线性方程组:

$$\begin{aligned} ah_0^2 + bh_0 = f_0 - c = e_0. \\ ah_1^2 + bh_1 = f_1 - c = e_1. \end{aligned} \tag{14}$$

求解线性方程组可得:

$$\begin{aligned} a &= \frac{e_0 h_1 - e_1 h_0}{h_1 h_0^2 - h_0 h_1^2} \\ b &= \frac{e_1 h_0^2 - e_0 h_1^2}{h_1 h_0^2 - h_0 h_1^2} \end{aligned} \tag{15}$$

下列二次式用来求解式(11)的根 $t = z_1, z_2$

$$Z = \frac{-2c}{b \pm \sqrt{b^2 - 4ac}} \tag{16}$$

式(16)等价于求二次根的标准公式,而且由于 $c = f_2$,所以它的情况更好。

为了确保方法的稳定性,需要选择式(16)中绝对值最小的根。如果 $b > 0$,使用带正号的根;如果 $b < 0$,使用带负号的根。则 p_3 如图 2.17 所示,表示为:

$$p_3 = p_2 + z \tag{17}$$

为了更新迭代,需要从 $\{p_0, p_1, p_2\}$ 中选择最靠近 p_3 的两点为新的 p_0 和 p_1(即放弃离 p_3 最远的一点)。然后使用新的 p_2 替代 p_3。尽管在 Muller 法中有许多辅助计算,但它每个迭代只需要计算一个函数。

如果利用 Muller 法求方程 $f(x) = 0$ 的实数根,可能会碰到复数近似值,因为式(16)中的平方根可能是复数(虚部不为零)。在这些情况下,虚部很小,可设为零,以便只需对实数进行计算。

2.5.3 方法之间的比较

Steffensen 法可与牛顿拉夫申固定点函数 $g(x) = x - f(x)/f'(x)$ 一起使用。在下面的两个例子中,对多项式 $f(x) = x^3 - 3x + 2$ 求根。牛顿拉夫申函数是 $g(x) = (2x^3 - 2)/(3x^2 - 3)$。当在程序 2.7 中使用这个函数,可得到在表 2.12 和表 2.13 中"结合牛顿法的 Steffensen 法"标题下的计算结果。例如,从 $p_0 = -2.4$ 开始,通过计算可得:

表 2.12　比较靠近单根处不同方法的收敛性

k	割线法	Muller 法	牛顿法	结合牛顿法的 Steffensen 法
0	-2.600000000	-2.600000000	-2.400000000	-2.400000000
1	-2.400000000	-2.500000000	-2.076190476	-2.076190476
2	-2.106598985	-2.400000000	-2.003596011	-2.003596011
3	-2.022641412	-1.985275287	-2.000008589	-1.982618143
4	-2.001511098	-2.000334062	-2.000000000	-2.000204982
5	-2.000022537	-2.000000218		-2.000000028
6	-2.000000022	-2.000000000		-2.000002389
7	-2.000000000			-2.000000000

$$p_1 = g(p_0) = -2.076190476 \tag{18}$$

$$p_2 = g(p_1) = -2.003596011 \tag{19}$$

则通过 Aitken 法可得 $p_3 = -1.982618143$。

例 2.19(靠近单根处的收敛性)　当函数 $f(x)=x^3-3x+2$ 靠近单根 $p=-2$ 处,对各种方法进行比较。

求此函数根的牛顿法和割线法,分别如例 2.14 和例 2.16 所示。表 2.12 对不同方法的计算结果进行了总结。

例 2.20(靠近二重根处的收敛性)　当函数 $f(x)=x^3-3x+2$ 靠近二重根 $p=1$ 处,对各种方法进行比较。表 2.13 给出来各种计算结果的汇总。

牛顿法是求解单根的最好选择(如表 2.12 所示)。对于二重根情况,Muller 法或结合牛顿拉夫申公式的 Steffensen 法是好的选择(如表 2.13 所示)。需要注意当序列 $\{p_k\}$ 收敛时,在 Aitken 加速式(4)中可能发生被零除的情况。在这种情况下,最后计算出的近似值可作为函数零点的近似值。

表 2.13　比较靠近二重根处不同方法的收敛性

k	割线法	Muller 法	牛顿法	结合牛顿法的 Steffensen 法
0	1.400000000	1.400000000	1.200000000	1.200000000
1	1.200000000	1.300000000	1.103030303	1.103030303
2	1.138461538	1.200000000	1.052356417	1.052356417
3	1.083873738	1.003076923	1.026400814	0.996890433
4	1.053093854	1.003838922	1.013257734	0.998446023
5	1.032853156	1.000027140	1.006643418	0.999223213
6	1.020429426	0.999997914	1.003325375	0.999999193
7	1.012648627	0.999999747	1.001663607	0.999999597
8	1.007832124	1.000000000	1.000832034	0.999999798
9	1.004844757		1.000416075	0.999999999
	⋮		⋮	

在下面的程序中,由结合牛顿拉夫申公式的 Steffensen 法生成的序列 $\{p_k\}$ 存储在矩阵 Q 中,矩阵 Q 有 max 行和 3 列。Q 的第一列包含根的初始近似值 p_0 它由 Aitken 加速法(公式(4))生成的项 $p_3, p_6, \cdots, p_{3k}, \cdots$。$Q$ 的第二列和第三列包含由牛顿法生成的项。程序的停止

判别条件是基于 Q 的第一列中的连续项的差值。

程序 2.7(Steffensen 加速法) 给定初始近似值 p_0,快速寻找固定点方程 $x = g(x)$ 的解,假设 $g(x)$ 和 $g'(x)$ 是连续的,而且 $|g'(x)| < 1$,通常的固定点迭代缓慢(线性)收敛到 p

```
function [P,Q] = steff(f,df,p0,delta,epsilon,max1)
% Input    - f is the object function input as a string'f'
%          - df is the derivative of f input as a string 'df'
%          - p0 is the initial approximation to a zero of f
%          - delta is the tolerance for p0
%          - epsilon is the tolerance for the function values y
%          - max1 is the maximum number of iterations
% Output   - p is the Steffensen approximation to the zero
%          - Q is the matrix containing the Steffensen sequence
% Initialize the matrix R
R = zeros(max1,3);
R(1,1) = P0;
for k = 1:max1
  for j = 2:3
     % Denominator in Newton - Raphson method is calculated
     nrodenom = feval(df,R(k,j - 1));
     % Calculate Newton - Raphson approximations
     if nrdenom = = 0
        'division by zero in Newton - Raphson method'
        break
     else
       R(k,j) = R(k,j - 1) - feval(f,R(k,j - 1))/nrdenom;
     end
     % Denominator in Aitken's Acceleration process calculated
     aadenom = R(k,3) - 2 * R(k,2) + R(k,1);
     % Calculate Aitken's Acceleration approximations
     if aadenom = = 0
       'division by zero in Aitken's Acceleration'
       break
     else
       R(k + 1,1) = R(k,1) - (R(k,2) - R(k,1))^2/aadenom;
     end
  end
  % End program if division by zero occurred
  if (nrdenom = = 0)|(aadenom = = 0)
     break
  end
  % Stopping criteria are evaluated
  err = abs(R(k,1) - R(k + 1,1));
  relerr = err/(abs(R(k + 1,1)) + delta);
  y = feval(f,R(k + 1,1));
  if (err < delta)|(relerr < delta)|(y < epsilon)
     % p and the matrix Q are determined
```

```
      p=R(k+1,1);
      Q=R(1:k+1,:);
      break
    end
  end
end
```

程序 2.8(Muler 法)　给定三个初始近似值 p_0,p_1 和 p_2,求方程 $f(x)=0$ 的根

```
function [p,y,err]=muller(f,p0,p1,p2,delta epsilon,max1)
%Input      - f is the object function input as a string 'f'
%           - p0,p1,and p2 are the initial approximations
%           - delta is the tolerance for p0,p1,and p2
%           - epsilon the the tolerance for the function values y
%           - max1 is the maximum number of iterations
%Output     - p is the Muller approximation to the zero of f
%           - y is the function value y=f(p)
%           - err is the error in the appoximation of p.
%Initialize the matrices P and Y
P=[p0 p1 p2];
Y=feval(f,P);
%Calculate a and b in formula (15)
for k=1:max1
    h0=P(1)-P(3);h1=P(2)-P(3);eo=Y(1)-Y(3);e1=Y(2)-Y(3);c=Y(3);
    denom=h1*h0^2-h0*h1^2;
    a=(e0*h1-e1*h0)/denom;
    b=(e1*h0^2-e0*h1^2)/denom;
    %Suppress any complex roots
    if b^2-4*a*c > 0
       disc=sqrt(b^2-4*a*c);
    else
       disc=0;
    end
    %Find the smallest root of(17)
    if b < 0
       disc=-disc;
    end
    z=-2*c/(b+disc);
    p=P(3)+z;
    %Sort the entries of P to find the two closest to p
    if abs(p-P(2))<abs(p-P(1))
       Q=[P(2) P(1) P(3)];
       P=Q;
       Y=feval(f,P);
    end
    if abs(p-P(3))<abs(p-P(2))
       R=[P(1) P(3) P(2)];
       P=R;
    Y=feval(f,P)
    end
    %Replace the entry of P that was farthest from P with p
```

```
    P(3)=p;
    Y(3)=feval(f,p(3));
    y=Y(3);
    % Determine stopping criteria
    err=abs(z);
    relerr=err/(abs(p)+delta);
    if (err<delta)|*relerr<delta|(abs(y)<epsilon)
       break
    end
  end
```

2.5.4 Aitken 法、Steffensen 法和 Muller 法的练习

1. 求 Δp_n,其中:

   (a) $p_n=5$ (b) $p_n=6n+2$ (c) $p_n=n(n+1)$

2. 设 $p_n=2n^2+1$,求 $\Delta^k p_n$,其中:

   (a) $k=2$ (b) $k=3$ (c) $k=4$

3. 设 $p_n=1/2^n$,证明对所有 n 有 $q_n=0$,其中 q_n 由式(4)给出。
4. 设 $p_n=1/n$,证明对所有 n 有 $q_n=1/(2n+2)$,而且收敛性基本没有加速。是否 $\{p_n\}$ 线性收敛到 0? 为什么?
5. 设 $p_n=1/(2^n-1)$,证明对所有 n,有 $q_n=1/(4^{n+1}-1)$。
6. 序列 $p_n=1/(4^n+4^{-n})$ 线性收敛到 0。利用 Aitken 公式(4)求 q_1,q_2 和 q_3,从而加速收敛。

n	p_n	q_n
0	0.5	-0.26437542
1	0.23529412	
2	0.06225681	
3	0.01562119	
4	0.00390619	
5	0.00097656	

7. 从 $p_0=2.5$ 开始,使用函数 $g(x)=(6+x)^{1/2}$,由固定点迭代生成的序列 $\{p_n\}$ 线性收敛到 $p=3$。利用 Aitken 公式(4)求解 q_1,q_2 和 q_3,从而加速收敛。
8. 从 $p_0=3.14$ 开始,使用函数 $g(x)=\ln(x)+2$,由固定点迭代生成的序列 $\{p_n\}$ 线性收敛到 $p\approx3.1419322$。利用 Aitken 公式(4)求解 q_1,q_2 和 q_3,从而加速收敛。
9. 对于方程 $\cos(x)-1=0$,牛顿拉夫申函数是 $g(x)=x-(1-\cos(x))/\sin(x)=x-\tan(x/2)$。从 $p_0=0.5$ 开始,利用结合 $g(x)$ 的 Steffensen 法,求解 p_1,p_2 和 p_3,再求解 p_4,p_5 和 p_6。
10. Aitken 法可用来加速序列的收敛性。如果序列的第 n 个部分和是:

$$S_n=\sum_{k=1}^{n}A_k$$

证明利用 Aitken 法导出的序列为:

$$T_n=S_n+\frac{A_{n+1}^2}{A_{n+1}-A_{n+2}}$$

在练习 11 到练习 14 中，利用 Aitken 法和练习 10 的结果加速序列的收敛。

11. $S_n = \sum_{k=1}^{n}(0.99)^k$

12. $S_n = \sum_{k=1}^{n}\dfrac{1}{4^k+4^{-k}}$

13. $S_n = \sum_{k=1}^{n}\dfrac{k}{2^{k-1}}$

14. $S_n = \sum_{k=1}^{n}\dfrac{1}{2^k k}$

15. 利用 Muller 法求方程 $f(x)=x^3-x-2$ 的根。从 $p_0=1.0, p_1=1.2$ 和 $p_2=1.4$ 开始，求 p_3, p_4 和 p_5。

16. 利用 Muller 法求方程 $f(x)=4x^2-e^x$ 的根。从 $p_0=4.0, p_1=4.1$ 和 $p_2=4.2$ 开始，求 p_3, p_4 和 p_5。

17. 设 $\{p_n\}$ 和 $\{q_n\}$ 是任意两个实数序列。证明：

    (a) $\Delta(p_n+q_n)=\Delta p_n+\Delta q_n$

    (b) $\Delta(p_n q_n)=p_{n+1}\Delta q_n+q_n\Delta p_n$

18. 对式(8)右边增加项 p_{n+2} 和 $-p_{n+2}$，证明下列公式与其等价。

$$p \approx p_{n+2}-\frac{(p_{n+2}-p_{n+1})^2}{p_{n+2}-2p_{n+1}+p_n}=q_n$$

19. 设迭代过程中的误差满足关系式 $E_{n+1}=KE_n$，其中 K 是某个常量，而且 $|K|<1$。

    (a) 求 E_n 的表达式，其中包含 E_0, K 和 n。

    (b) 求最小整数 N 的表达式，满足 $|E_N|<10^{-8}$。

2.5.5 算法和程序

1. 利用 Steffensen 法，初始近似值 $p_0=0.5$，求解 $f(x)=x-\sin(x)$ 的零点近似值，精确到小数点后 10 位。
2. 利用 Steffensen 法，初始近似值 $p_0=0.5$，求解 $f(x)=\sin(x^3)$ 的最接近 0.5 的零点近似值，精确到小数点后 10 位。
3. 利用 Muller 法，初始近似值 $p_0=1.5, p_1=1.4, p_2=1.3$，求解 $f(x)=1+2x-\tan(x)$ 的零点，精确到小数点后 12 位。
4. 在程序 2.8(Muller 法)中，用 p_0, p_1 和 p_2 初始化一个 1×3 矩阵 $\boldsymbol{P}$。在循环结尾处，p_0，p_1 或 p_2 中的一个由新的零点近似值替换。这个过程一直持续到终止评定条件 $k=K$ 满足为止。修改程序 2.8，使得除了 p 和 err 外，还将产生一个 $(K+1)\times3$ 矩阵 $\boldsymbol{Q}$，$\boldsymbol{Q}$ 的第一行包含 1×3 矩阵 $\boldsymbol{P}$，$\boldsymbol{P}$ 的初始值为零点的近似值。$\boldsymbol{Q}$ 的第 k 行包含由三个零点近似值构成的第 k 个集合。

   使用修改过的程序 2.8 和初始近似值 $p_0=2.4, p_1=2.3, p_2=2.2$，求解 $f(x)=3\cos(x)+2\sin(x)$ 的零点，精确到小数点后 8 位。

第 3 章　线性方程组 $AX = B$ 的数值解法

在第一象限(octant)内有阴影表示的三个平面(plane),如图 3.1 所示。设三个平面的方程为:

$$5x + y + z = 5$$
$$x + 4y + z = 4$$
$$x + y + 3z = 3$$

图 3.1　三个平面的交点

三个平面的交点坐标是什么？可以用高斯消去法,求出上述线性方程组的解:

$$x = 0.76,\quad y = 0.68 \text{ 和 } z = 0.52$$

在这一章中,将主要研究求解线性方程组的数值解法。

3.1　向量和矩阵介绍

一个 N 维实数向量 X 是 N 个实数的有序集合,通常写成坐标形式:

$$X = (x_1, x_2, \cdots, x_N) \tag{1}$$

这里数 $x_1, x_2, \cdots$,和 x_N 称为 X 的坐标或分量。包含所有 N 维向量的集合,称为 N 维空间。当一个向量用来表示空间中的一点或位置时,称之为位置向量(***position vector***);当用它来表示空间两点的移动时,则称之为偏移向量(***displacement vector***)。

设另一个向量为 $Y = (y_1, y_2, \cdots, y_N)$。当且仅当它们对应的分量相等时两个向量 X 和 Y 相等,表示为:

$$X = Y \quad \text{当且仅当 } x_j = y_j, j = 1, 2, \cdots, N \tag{2}$$

向量 X 与 Y 的和是它们的对应分量分别相加得到的向量，表示为：

$$X+Y=(x_1+y_1, x_2+y_2, \cdots, x_N+y_N) \tag{3}$$

可通过将它的所有分量取负对向量 X 取负，表示为：

$$-X=(-x_1, -x_2, \cdots, -x_N) \tag{4}$$

可通过将它们的对应分量相减得到 Y 与 X 的差，表示为：

$$Y-X=(y_1-x_1, y_2-x_2, \cdots, y_N-x_N) \tag{5}$$

N 维空间服从代数性质：

$$Y-X=Y+(-X) \tag{6}$$

如果 c 是实数(标量)，则定义标量乘积 cX 为：

$$cX=(cx_1, cx_2, \cdots, cx_N) \tag{7}$$

如果 c 和 d 是标量，则加权和 $cX+dY$ 称为 X 和 Y 的线性组合，表示为：

$$cX+dY=(cx_1+dy_1, cx_2+dy_2, \cdots, cx_N+dy_N) \tag{8}$$

X 和 Y 的点积是一个标量值(实数)，定义为：

$$X\cdot Y=x_1y_1+x_2y_2+\cdots+x_Ny_N \tag{9}$$

向量 X 的模(长度)定义为：

$$\|X\|=(x_1^2+x_2^2+\cdots+x_N^2)^{1/2} \tag{10}$$

式(10)称为向量 X 的欧几里德模(长度)。

当 $|c|>1$ 时，标量乘积 cX 拉伸向量 X；当 $|c|<1$ 时，标量乘积 cX 压缩向量 X。这可通过式(10)看出：

$$\begin{aligned}\|cX\| &= (c^2x_1^2+c^2x_2^2+\cdots+c^2x_N^2)^{1/2} \\ &= |c|(x_1^2+x_2^2+\cdots+x_N^2)^{1/2}=|c|\ \|X\|\end{aligned} \tag{11}$$

在点积和向量的模之间存在一个重要的关系。如果对式(10)两边平方，并在式(9)中用 X 替换 Y，可得：

$$\|X\|^2=x_1^2+x_2^2+\cdots+x_N^2=X\cdot X \tag{12}$$

如果 X 和 Y 是位置向量，用位于 N 维空间中的点 $(x_1, x_2, \cdots, x_N)$ 和 $(y_1, y_2, \cdots, y_N)$ 表示，则从 X 到 Y 的偏移向量是它们的差：

$$Y-X \text{(从位置 } X \text{ 到位置 } Y \text{ 的偏移)} \tag{13}$$

注意如果一个粒子从位置 X 开始，沿偏移 $Y-X$ 移动，则它的新位置为 Y。这可通过如下的向量和得到：

$$Y=X+(Y-X) \tag{14}$$

利用式(10)和式(13)，可得到 N 维空间中两点的距离公式：

$$\|Y-X\|=((y_1-x_1)^2+(y_2-x_2)^2+\cdots+(y_N-x_N)^2)^{1/2} \tag{15}$$

当用式(15)计算两点间的距离时，称这些点位于 N 维欧几里德空间。

例 3.1　设 $X=(2,-3,5,-1)$ 和 $Y=(6,1,2,-4)$。上述概念可用下列 4 维空间的向量计算表示：

和　　$X+Y=(8,-2,7,-5)$

差　　$X-Y=(-4,-4,3,3)$

标量乘	$3\boldsymbol{X}=(6,-9,15,-3)$
长度	$\|\boldsymbol{X}\|=(4+9+25+1)^{1/2}=39^{1/2}$
点积	$\boldsymbol{X}\cdot\boldsymbol{Y}=12-3+10+4=23$
从 $\boldsymbol{X}$ 到 $\boldsymbol{Y}$ 的偏移	$\boldsymbol{Y}-\boldsymbol{X}=(4,4,-3,-3)$
从 $\boldsymbol{X}$ 到 $\boldsymbol{Y}$ 的距离	$\|\boldsymbol{Y}-\boldsymbol{X}\|=(16+16+9+9)^{1/2}=50^{1/2}$

有时将行向量写成列向量更有用,例如:

$$\boldsymbol{X}=\begin{bmatrix}x_1\\x_2\\\vdots\\x_N\end{bmatrix} \quad 和 \quad \boldsymbol{Y}=\begin{bmatrix}y_1\\y_2\\\vdots\\y_N\end{bmatrix} \tag{16}$$

则线性组合 $c\boldsymbol{X}+d\boldsymbol{Y}$ 可表示为:

$$c\boldsymbol{X}+d\boldsymbol{Y}=\begin{bmatrix}cx_1+dy_1\\cx_2+dy_2\\\vdots\\cx_N+dy_N\end{bmatrix} \tag{17}$$

在式(17)中,通过适当选择 c 和 d,可得到和 $1\boldsymbol{X}+1\boldsymbol{Y}$、差 $1\boldsymbol{X}-1\boldsymbol{Y}$ 和标量乘积 $c\boldsymbol{X}+0\boldsymbol{Y}$。使用上标"′"表示向量的转置,即行向量变成列向量,或反之:

$$(x_1,x_2,\cdots,x_N)'=\begin{bmatrix}x_1\\x_2\\\vdots\\x_N\end{bmatrix} \quad 和 \quad \begin{bmatrix}x_1\\x_2\\\vdots\\x_N\end{bmatrix}'=(x_1,x_2,\cdots,x_N) \tag{18}$$

向量的集合有零元素 $\boldsymbol{0}$,定义为:

$$\boldsymbol{0}=(0,0,\cdots,0) \tag{19}$$

定理 3.1(向量代数) 设 $\boldsymbol{X}$, $\boldsymbol{Y}$ 和 $\boldsymbol{Z}$ 是 N 维向量,a 和 b 是标量(实数)。向量加和标量乘积有如下性质:

$$\boldsymbol{Y}+\boldsymbol{X}=\boldsymbol{X}+\boldsymbol{Y} \quad 交换律 \tag{20}$$

$$\boldsymbol{0}+\boldsymbol{X}=\boldsymbol{X}+\boldsymbol{0} \quad 加法单位元 \tag{21}$$

$$\boldsymbol{X}-\boldsymbol{X}=\boldsymbol{X}+(-\boldsymbol{X})=\boldsymbol{0} \quad 加法逆元 \tag{22}$$

$$(\boldsymbol{X}+\boldsymbol{Y})+\boldsymbol{Z}=\boldsymbol{X}+(\boldsymbol{Y}+\boldsymbol{Z}) \quad 结合律 \tag{23}$$

$$(a+b)\boldsymbol{X}=a\boldsymbol{X}+b\boldsymbol{X} \quad 标量分配律 \tag{24}$$

$$a(\boldsymbol{X}+\boldsymbol{Y})=a\boldsymbol{X}+a\boldsymbol{Y} \quad 向量分配律 \tag{25}$$

$$a(b\boldsymbol{X})=(ab)\boldsymbol{X} \quad 标量结合律 \tag{26}$$

3.1.1 矩阵和二维数组

一个矩阵是数字按行列分布的矩形数组。一个矩阵有 M 行和 N 列,称为 $M\times N$ 矩阵。大写字母 $\boldsymbol{A}$ 表示一个矩阵,小写带下标字母 a_{ij} 表示构成矩阵的一个数。矩阵可表示为:

$$\boldsymbol{A}=[a_{ij}]_{M\times N}, \quad 1\leqslant i\leqslant M, 1\leqslant j\leqslant N \tag{27}$$

这里 a_{ij} 是位于(i,j)(即存储在矩阵的第 i 行和第 j 列上)的数。将 a_{ij} 作为位于(i,j)的元素。其扩展形式为：

$$\text{行 } i \rightarrow \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1j} & \cdots & a_{1N} \\ a_{21} & a_{22} & \cdots & a_{2j} & \cdots & a_{2N} \\ \vdots & \vdots & & \vdots & & \vdots \\ a_{i1} & a_{i2} & \cdots & a_{ij} & \cdots & a_{iN} \\ \vdots & \vdots & & \vdots & & \vdots \\ a_{M1} & a_{M2} & \cdots & a_{Mj} & \cdots & a_{MN} \end{bmatrix} = A \tag{28}$$

$$\uparrow \text{ 列 } j$$

$M \times N$ 矩阵 A 的行是 N 维向量：

$$V_i = (a_{i1}, a_{i2}, \cdots, a_{iN}), \qquad i = 1,2,\cdots,M \tag{29}$$

式(29)中的行向量也可看成 $1 \times N$ 矩阵。将 $M \times N$ 矩阵 A 分解成 M 块(子矩阵)即形成 $1 \times N$ 矩阵。

在这种情况下，可将 A 表示为一个 $M \times 1$ 矩阵，而该矩阵由 $1 \times N$ 矩阵 V_i 组成，即：

$$A = \begin{bmatrix} V_1 \\ V_2 \\ \vdots \\ V_i \\ \vdots \\ V_M \end{bmatrix} = [V_1 \quad V_2 \cdots V_i \cdots V_M]' \tag{30}$$

同理，$M \times N$ 矩阵 A 的列形成 $M \times 1$ 矩阵：

$$C_1 = \begin{bmatrix} a_{11} \\ a_{21} \\ \vdots \\ a_{i1} \\ \vdots \\ a_{M1} \end{bmatrix}, \cdots, C_j = \begin{bmatrix} a_{1j} \\ a_{2j} \\ \vdots \\ a_{ij} \\ \vdots \\ a_{Mj} \end{bmatrix}, \cdots, C_N = \begin{bmatrix} a_{1N} \\ a_{2N} \\ \vdots \\ a_{iN} \\ \vdots \\ a_{MN} \end{bmatrix} \tag{31}$$

在这种情况下，可将 A 表示成由 $1 \times N$ 列矩阵，而 C_j 构成的 $M \times 1$ 矩阵：

$$A = [C_1 \quad C_2 \cdots C_j \cdots C_N] \tag{32}$$

例 3.2　标识下列 4×3 矩阵的行矩阵和列矩阵：

$$A = \begin{bmatrix} -2 & 4 & 9 \\ 5 & -7 & 1 \\ 0 & -3 & 8 \\ -4 & 6 & -5 \end{bmatrix}$$

解：

4 行矩阵为：$V_1 = [-2 \quad 4 \quad 9]$，$V_2 = [5 \quad -7 \quad 1]$，$V_3 = [0 \quad -3 \quad 8]$和 $V_4 = [-4 \quad 6 \quad -5]$。

3列矩阵为:

$$C_1=\begin{bmatrix}-2\\5\\0\\-4\end{bmatrix},\quad C_2=\begin{bmatrix}4\\-7\\-3\\6\end{bmatrix}\quad 和\quad C_3=\begin{bmatrix}9\\1\\8\\-5\end{bmatrix}$$

$\boldsymbol{A}$ 可用这些矩阵进行表示:

$$\boldsymbol{A}=\begin{bmatrix}\boldsymbol{V}_1\\\boldsymbol{V}_2\\\boldsymbol{V}_3\\\boldsymbol{V}_4\end{bmatrix}=[\boldsymbol{C}_1\quad\boldsymbol{C}_2\quad\boldsymbol{C}_3]$$

设 $\boldsymbol{A}=[a_{ij}]_{M\times N}$和$\boldsymbol{B}=[b_{ij}]_{M\times N}$为两个同维矩阵。当且仅当二者每个对应元素相等时 $\boldsymbol{A}$ 与$\boldsymbol{B}$ 相等,即:

$$\boldsymbol{A}=\boldsymbol{B}\ 当且仅当\ a_{ij}=b_{ij}\quad,\quad 1\leqslant i\leqslant M\quad 1\leqslant j\leqslant N \tag{33}$$

两个 $M\times N$ 矩阵$\boldsymbol{A}$ 与$\boldsymbol{B}$ 的和的定义如下:

$$\boldsymbol{A}+\boldsymbol{B}=[a_{ij}+b_{ij}]_{M\times N}\quad,\quad 1\leqslant i\leqslant M\quad 1\leqslant j\leqslant N \tag{34}$$

可通过对 $\boldsymbol{A}$ 中每个元素取负得到矩阵$\boldsymbol{A}$ 取负:

$$-\boldsymbol{A}=[-a_{ij}]_{M\times N}\quad,\quad 1\leqslant i\leqslant M\quad 1\leqslant j\leqslant N \tag{35}$$

可通过求对应分量的差得到 $\boldsymbol{A}-\boldsymbol{B}$:

$$\boldsymbol{A}-\boldsymbol{B}=[a_{ij}-b_{ij}]_{M\times N}\quad,\quad 1\leqslant i\leqslant M\quad 1\leqslant j\leqslant N \tag{36}$$

如果 c 是实数(标量),可定义标量乘积 $c\boldsymbol{A}$ 为:

$$c\boldsymbol{A}=[ca_{ij}]_{M\times N}\quad,\quad 1\leqslant i\leqslant M\quad 1\leqslant j\leqslant N \tag{37}$$

如果 p 和q 是标量,加权和 $p\boldsymbol{A}+q\boldsymbol{B}$ 称为矩阵$\boldsymbol{A}$ 和$\boldsymbol{B}$ 的线性组合,表示为:

$$p\boldsymbol{A}+q\boldsymbol{B}=[pa_{ij}+qb_{ij}]_{M\times N}\quad,\quad 1\leqslant i\leqslant M\quad 1\leqslant j\leqslant N \tag{38}$$

$M\times N$ 零矩阵由零元素构成:

$$\boldsymbol{0}=[0]_{M\times N} \tag{39}$$

例3.3 求矩阵乘 $2\boldsymbol{A}$ 和 $3\boldsymbol{B}$,以及线性组合 $2\boldsymbol{A}-3\boldsymbol{B}$,$\boldsymbol{A}$ 和 $\boldsymbol{B}$ 的值如下所示:

$$\boldsymbol{A}=\begin{bmatrix}-1&2\\7&5\\3&-4\end{bmatrix}\quad 和\quad \boldsymbol{B}=\begin{bmatrix}-2&3\\1&-4\\-9&7\end{bmatrix}$$

利用式(37),可得:

$$2\boldsymbol{A}=\begin{bmatrix}-2&4\\14&10\\6&-8\end{bmatrix}\quad 和\quad 3\boldsymbol{B}=\begin{bmatrix}-6&3\\3&-12\\-27&21\end{bmatrix}$$

线性组合 $2\boldsymbol{A}-3\boldsymbol{B}$ 为:

$$2\boldsymbol{A}-3\boldsymbol{B}=\begin{bmatrix}-2+6&4-9\\14-3&10+12\\6+27&-8-21\end{bmatrix}=\begin{bmatrix}4&-5\\11&22\\33&-29\end{bmatrix}$$

定理 3.2（矩阵加） 设 $\boldsymbol{A},\boldsymbol{B}$ 和 $\boldsymbol{C}$ 是 $M\times N$ 矩阵，p 和 q 为标量。矩阵加和标量乘积有如下性质：

$\boldsymbol{B}+\boldsymbol{A}=\boldsymbol{A}+\boldsymbol{B}$	交换律	(40)
$\boldsymbol{0}+\boldsymbol{A}=\boldsymbol{A}+\boldsymbol{0}$	加法单位元	(41)
$\boldsymbol{A}-\boldsymbol{A}=\boldsymbol{A}+(-\boldsymbol{A})=\boldsymbol{0}$	加法逆元	(42)
$(\boldsymbol{A}+\boldsymbol{B})+\boldsymbol{C}=\boldsymbol{A}+(\boldsymbol{B}+\boldsymbol{C})$	结合律	(43)
$(p+q)\boldsymbol{A}=p\boldsymbol{A}+q\boldsymbol{A}$	标量分配律	(44)
$p(\boldsymbol{A}+\boldsymbol{B})=p\boldsymbol{A}+p\boldsymbol{B}$	矩阵分配律	(45)
$p(q\boldsymbol{A})=(pq)\boldsymbol{A}$	标量结合律	(46)

3.1.2 向量和矩阵简介的练习

最好手工计算或用 MATLAB 完成下述练习。

1. 给定向量 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$，求解(a) $\boldsymbol{X}+\boldsymbol{Y}$，(b) $\boldsymbol{X}-\boldsymbol{Y}$，(c) $3\boldsymbol{X}$，(d) $\|\boldsymbol{X}\|$，(e) $7\boldsymbol{Y}-4\boldsymbol{X}$，(f) $\boldsymbol{X}\cdot\boldsymbol{Y}$ 及(g) $\|7\boldsymbol{Y}-4\boldsymbol{X}\|$。
   (i) $\boldsymbol{X}=(3,-4)$ 和 $\boldsymbol{Y}=(-2,8)$
   (ii) $\boldsymbol{X}=(-6,3,2)$ 和 $\boldsymbol{Y}=(-8,5,1)$
   (iii) $\boldsymbol{X}=(4,-8,1)$ 和 $\boldsymbol{Y}=(1,-12,-11)$
   (iv) $\boldsymbol{X}=(1,-2,4,2)$和 $\boldsymbol{Y}=(3,-5,-4,0)$
2. 利用余弦定律，证明两个向量 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 之间的角度 θ 可用下述关系式表示：
   $$\cos(\theta)=\frac{\boldsymbol{X}\cdot\boldsymbol{Y}}{\|\boldsymbol{X}\|\ \|\boldsymbol{Y}\|}$$
   针对下列向量，求向量间的角度(用弧度单位)：
   (a) $\boldsymbol{X}=(-6,3,2)$ 和 $\boldsymbol{Y}=(2,-2,1)$
   (b) $\boldsymbol{X}=(4,-8,1)$ 和 $\boldsymbol{Y}=(3,4,12)$
3. 如果两个向量 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 之间的角度为 $\pi/2$，则称这两个向量正交(垂直)。
   (a) 证明 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 正交当且仅当 $\boldsymbol{X}\cdot\boldsymbol{Y}=0$。
   利用(a) 的结论判定下列向量是否正交。
   (b) $\boldsymbol{X}=(-6,4,2)$ 和 $\boldsymbol{Y}=(6,5,8)$
   (c) $\boldsymbol{X}=(-4,8,3)$ 和 $\boldsymbol{Y}=(2,5,16)$
   (d) $\boldsymbol{X}=(-5,7,2)$ 和 $\boldsymbol{Y}=(4,1,6)$
   (e) 求与向量 $\boldsymbol{X}=(1,2,-5)$正交的两个不同向量。
4. 对下列矩阵，求解(a) $\boldsymbol{A}+\boldsymbol{B}$，(b) $\boldsymbol{A}-\boldsymbol{B}$ 及(c) $3\boldsymbol{A}-2\boldsymbol{B}$。矩阵表示如下：
   $$\boldsymbol{A}=\begin{bmatrix}-1 & 9 & 4\\ 2 & -3 & -6\\ 0 & 5 & 7\end{bmatrix},\quad \boldsymbol{B}=\begin{bmatrix}-4 & 9 & 2\\ 3 & -5 & 7\\ 8 & 1 & -6\end{bmatrix}$$
5. $M\times N$ 矩阵$\boldsymbol{A}$ 的转置表示为$\boldsymbol{A}'$，通过将 $\boldsymbol{A}$ 的行变成$\boldsymbol{A}'$的列构成$N\times M$ 矩阵。也就是说，如果 $\boldsymbol{A}=[a_{ij}]_{M\times N}$且$\boldsymbol{A}'=[b_{ij}]_{N\times M}$，则相应元素满足关系式：
   $$b_{ji}=a_{ij}\quad,\quad 1\leqslant i\leqslant M,1\leqslant j\leqslant N$$
   求下列矩阵的转置：

(a) $\begin{bmatrix} -2 & 5 & 12 \\ 1 & 4 & -1 \\ 7 & 0 & 6 \\ 11 & -3 & 8 \end{bmatrix}$ (b) $\begin{bmatrix} 4 & 9 & 2 \\ 3 & 5 & 7 \\ 8 & 1 & 6 \end{bmatrix}$

6. 如果 $N \times N$ 方阵 $\boldsymbol{A}$ 满足 $\boldsymbol{A}=\boldsymbol{A}'$(参见练习 5 中对 $\boldsymbol{A}'$ 的定义),则称 $\boldsymbol{A}$ 为对称矩阵。判断下列方阵是否对称:

(a) $\begin{bmatrix} 1 & -7 & 4 \\ -7 & 3 & 0 \\ 4 & 0 & 3 \end{bmatrix}$ (b) $\begin{bmatrix} 4 & -7 & 1 \\ 0 & 2 & -7 \\ 3 & 0 & 4 \end{bmatrix}$

(c) $\boldsymbol{A}=[a_{ij}]_{N\times N}$, 其中 $a_{ij}=\begin{cases} ij & i=j \\ i-ij+j & i\neq j \end{cases}$

(d) $\boldsymbol{A}=[a_{ij}]_{N\times N}$, 其中 $a_{ij}=\begin{cases} \cos(ij) & i=j \\ i-ij-j & i\neq j \end{cases}$

7. 证明定理 3.1 中的命题(20)、(24)和(25)。

3.2 向量和矩阵的性质

变量 $x_1, x_2, \cdots, x_N$ 的线性组合为:

$$a_1x_1 + a_2x_2 + \cdots + a_Nx_N \tag{1}$$

其中 a_k 是 x_k 的系数,$k=1,2,\cdots,N$。

赋给线性组合(1)一个定值 b,可得到一个关于 $x_1,x_2,\cdots,x_N$ 的线性方程,表示为:

$$a_1x_1 + a_2x_2 + \cdots + a_Nx_N = b \tag{2}$$

线性方程组经常出现,而且如果有 M 个方程,N 个未知数,则它可表示为:

$$\begin{array}{ccccc} a_{11}x_1 & + a_{12}x_2 & + \cdots + & a_{1N}x_N & = b_1 \\ a_{21}x_1 & + a_{22}x_2 & + \cdots + & a_{2N}x_N & = b_2 \\ \vdots & \vdots & & \vdots & \vdots \\ a_{k1}x_1 & + a_{k2}x_2 & + \cdots + & a_{kN}x_N & = b_k \\ \vdots & \vdots & & \vdots & \vdots \\ a_{M1}x_1 & + a_{M2}x_2 & + \cdots + & a_{MN}x_N & = b_M \end{array} \tag{3}$$

为了清楚表示每个方程的不同系数,必须使用两个下标(k,j)。第一个下标定位方程 k,第二个下标定位变量 x_j。

式(3)的解是同时满足式(3)中所有方程的一组数值 $x_1,x_2,\cdots,x_N$。因此可将解表示为 N 维向量:

$$\boldsymbol{X}=(x_1,x_2,\cdots,x_N) \tag{4}$$

例 3.4 混凝土是水泥、沙子和砂砾构成的混合物。发行商有三批选择给承包人。第一批包含水泥、沙子和砂砾的比例为 1/8、3/8、4/8;第二批的比例是 2/10、5/10、3/10;第三批的比例是2/5、3/5、0/5。

设 x_1、x_2、x_3 表示每批混合物的数量(单位为立方英码),它们的总量为 10 立方英码。水泥、沙子和砂砾的数量分别为 $b_1 = 2.3, b_2 = 4.8, b_3 = 2.9$。则关于这三种成分的线性方程组为:

$$
\begin{aligned}
0.125x_1 + 0.200x_2 + 0.400x_3 &= 2.3 \quad (\text{水泥}) \\
0.375x_1 + 0.500x_2 + 0.600x_3 &= 4.8 \quad (\text{沙子}) \\
0.500x_1 + 0.300x_2 + 0.000x_3 &= 2.9 \quad (\text{砂砾})
\end{aligned} \tag{5}
$$

线性方程组(5)的解为 $x_1 = 4, x_2 = -3, x_3 = 3$。可将它们直接代入方程组中进行检验:

$$
\begin{aligned}
(0.125)(4) + (0.200)(3) + (0.400)(3) &= 2.3 \\
(0.375)(4) + (0.500)(3) + (0.600)(3) &= 4.8 \\
(0.500)(4) + (0.300)(3) + (0.000)(3) &= 2.9
\end{aligned}
$$

3.2.1 矩阵乘

定义 3.1 如果 $A = [a_{ik}]_{M\times N}$, $B = [b_{kj}]_{N\times P}$, A 的列数和 B 的行数相等,则矩阵乘积 AB 为 $M \times P$ 的矩阵 C:

$$AB = C = [c_{ij}]_{M\times P} \tag{6}$$

其中 C 的元素 c_{ij} 是 A 的第 i 行与 B 的第 j 列的点积:

$$c_{ij} = \sum_{k=1}^{N} a_{ik}b_{kj} = a_{i1}b_{1j} + a_{i2}b_{2j} + \cdots + a_{iN}b_{Nj} \tag{7}$$

其中 $i = 1,2,\cdots,M$ 且 $j = 1,2,\cdots P$。

例 3.5 使用下列矩阵求解矩阵乘积 $C = AB$,说明为何 BA 无定义:

$$A = \begin{bmatrix} 2 & 3 \\ -1 & 4 \end{bmatrix}, \qquad B = \begin{bmatrix} 5 & -2 & 1 \\ 3 & 8 & -6 \end{bmatrix}$$

解:

矩阵 A 有两列,矩阵 B 有两行,所以矩阵乘积 AB 有定义。2×2 矩阵和 2×3 矩阵的乘积是 2×3 矩阵。计算过程如下:

$$
\begin{aligned}
AB &= \begin{bmatrix} 2 & 3 \\ -1 & 4 \end{bmatrix}\begin{bmatrix} 5 & -2 & 1 \\ 3 & 8 & -6 \end{bmatrix} \\
&= \begin{bmatrix} 10+9 & -4+24 & 2-18 \\ -5+12 & 2+32 & -1-24 \end{bmatrix} = \begin{bmatrix} 19 & 20 & -16 \\ 7 & 34 & -25 \end{bmatrix} = C
\end{aligned}
$$

当试图计算矩阵的乘积 BA 时,会发现二者不匹配。因为 B 的行是 3 维向量,而 A 的列是 2 维向量,因此 B 的第 j 行与 A 的第 k 列的点积没有定义。

如果 $AB = BA$,则称 A 与 B 可交换(commute)。而且通常 AB 与 BA 都有定义,但它们的乘积并不一样。

现在讨论怎样使用矩阵表示一个线性方程组。式(3)中的线性方程可写成一个矩阵乘积。系数 a_{kj} 保存在 $M\times N$ 矩阵 A(称为系数矩阵)中,而未知数 x_j 保存在 $N\times 1$ 矩阵 X 中。常数 b_k 存储在 $M\times 1$ 矩阵 B 中。根据惯例,使用列矩阵表示 X 和 B,则 AX 表示为:

$$\boldsymbol{AX}=\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1j} & \cdots & a_{1N} \\ a_{21} & a_{22} & \cdots & a_{2j} & \cdots & a_{2N} \\ \vdots & \vdots & & \vdots & & \vdots \\ a_{k1} & a_{k2} & \cdots & a_{kj} & \cdots & a_{kN} \\ \vdots & \vdots & & \vdots & & \vdots \\ a_{M1} & a_{M2} & \cdots & a_{Mj} & \cdots & a_{MN} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_j \\ \vdots \\ x_N \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_j \\ \vdots \\ b_M \end{bmatrix} = \boldsymbol{B} \tag{8}$$

式(8)中的矩阵乘 $\boldsymbol{AX}=\boldsymbol{B}$ 类似于通常向量的点积,因为 $\boldsymbol{B}$ 中的每个元素 b_k 都可通过矩阵 $\boldsymbol{A}$ 的第 k 行与列矩阵 $\boldsymbol{X}$ 的点积得到。

例 3.6 将例 3.4 中的线性方程组(5)用矩阵乘积表示。使用矩阵乘积验证[4 3 3]′是方程组(5)的解:

$$\begin{bmatrix} 0.125 & 0.200 & 0.400 \\ 0.375 & 0.500 & 0.600 \\ 0.500 & 0.300 & 0.000 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2.3 \\ 4.8 \\ 2.9 \end{bmatrix} \tag{9}$$

为了验证[4 3 3]′是方程组(5)的解,必须证明 $\boldsymbol{A}[4\ \ 3\ \ 3]'=[2.3\ \ 4.8\ \ 2.9]'$:

$$\begin{bmatrix} 0.125 & 0.200 & 0.400 \\ 0.375 & 0.500 & 0.600 \\ 0.500 & 0.300 & 0.000 \end{bmatrix} \begin{bmatrix} 4 \\ 3 \\ 3 \end{bmatrix} = \begin{bmatrix} 0.5+0.6+1.2 \\ 1.5+1.5+1.8 \\ 2.0+0.9+0.0 \end{bmatrix} = \begin{bmatrix} 2.3 \\ 4.8 \\ 2.9 \end{bmatrix}$$

3.2.2 特殊矩阵

元素全为零的 $M\times N$ 矩阵称为 $M\times N$ 维零矩阵,表示为:

$$\mathbf{0}=[0]_{M\times N} \tag{10}$$

当矩阵维数明确时,可使用 **0** 表示零矩阵。

N 阶单位矩阵是方阵,表示为:

$$\boldsymbol{I}_N=[\delta_{ij}]_{N\times M} \quad , \quad \delta_{ij}=\begin{cases} 1 & 当\ i=j \\ 0 & 当\ i\neq j \end{cases} \tag{11}$$

它是乘积单位矩阵,通过下例可以看出。

例 3.7 设 $\boldsymbol{A}$ 是 2×3 矩阵,则 $\boldsymbol{I}_2\boldsymbol{A}=\boldsymbol{AI}_2=\boldsymbol{A}$。$\boldsymbol{I}_2$ 左乘 $\boldsymbol{A}$ 的结果为:

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix} = \begin{bmatrix} a_{11}+0 & a_{12}+0 & a_{13}+0 \\ a_{21}+0 & a_{22}+0 & a_{23}+0 \end{bmatrix} = \boldsymbol{A}$$

$\boldsymbol{I}_3$ 右乘 $\boldsymbol{A}$ 的结果为:

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} a_{11}+0+0 & 0+a_{12}+0 & 0+0+a_{12} \\ a_{21}+0+0 & 0+a_{22}+0 & 0+0+a_{23} \end{bmatrix} = \boldsymbol{A}$$

下面的定理给出了关于矩阵乘的一些性质。

定理 3.3(矩阵乘) 设 c 是一个标量,$\boldsymbol{A},\boldsymbol{B},\boldsymbol{C}$ 是矩阵,而且对应的矩阵加法和乘法有定义,则:

$$(\boldsymbol{AB})\boldsymbol{C}=\boldsymbol{A}(\boldsymbol{BC}) \qquad 矩阵乘的结合律 \tag{12}$$

$IA = AI = A$	单位矩阵	(13)
$A(B + C) = AB + AC$	左分配律	(14)
$(A + B)C = AC + BC$	右分配律	(15)
$c(AB) = (cA)B = A(cB)$	标量结合律	(16)

3.2.3 非奇异矩阵的逆

矩阵逆的存在需要特殊条件。如果存在一个 $N \times N$ 矩阵 B 满足式(17)，则称 $N \times N$ 矩阵 A 是非奇异或可逆的：

$$AB = BA = I \tag{17}$$

如果矩阵 B 不存在，则称 A 是奇异矩阵。当存在 B 且满足式(17)，则称 B 是 A 的逆，通常表示为 $B = A^{-1}$，并使用相似的关系式：

$$AA^{-1} = A^{-1}A \quad \text{如果 } A \text{ 是非奇异矩阵} \tag{18}$$

显见至少存在一个矩阵 B 可满足关系式(17)。设 C 是 A 的一个逆(即 $AC = CA = I$)，则根据性质式(12)和式(13)有：

$$C = IC = (BA)C = B(AC) = BI = B$$

3.2.4 行列式

方阵 A 的行列式是一个标量值(实数)，表示为 $\det(A)$ 或 $|A|$。如果 A 是 $N \times N$ 矩阵：

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1N} \\ a_{21} & a_{22} & \cdots & a_{2N} \\ \vdots & \vdots & & \vdots \\ a_{N1} & a_{N2} & \cdots & a_{NN} \end{bmatrix}$$

则 A 的行列式表示为：

$$\det(A) = \begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1N} \\ a_{21} & a_{22} & \cdots & a_{2N} \\ \vdots & \vdots & & \vdots \\ a_{N1} & a_{N2} & \cdots & a_{NN} \end{vmatrix}$$

尽管行列式的表示类似于一个矩阵，但它的性质完全不同，行列式是一个标量值(实数)。当 $N > 3$ 时，根据大多数线性代数课本中定义的行列式 $\det(A)$，不易计算它的值。下面将回顾一下，如何利用代数余子式展开法计算行列式的值。计算高阶行列式可利用高斯消去法，可参见程序 3.3。

如果 $A = [a_{ij}]$ 是 1×1 矩阵，定义 $\det(A) = a_{11}$。如果 $A = [a_{ij}]_{N \times N}$，其中 $N \geqslant 2$，则让 M_{ij} 为 A 的 $N-1 \times N-1$ 子矩阵的行列式，子矩阵是通过划掉矩阵 A 的第 i 行和第 j 列构成的。行列式 M_{ij} 称为 a_{ij} 的余子式。A_{ij} 定义为 $A_{ij} = (-1)^{i+j}M_{ij}$，称为 a_{ij} 的代数余子式。这样 $N \times N$ 矩阵 A 的行列式表示为：

$$\det(A) = \sum_{j=1}^{N} a_{ij}A_{ij} \quad (\text{第 } i \text{ 行展开}) \tag{19}$$

或：

$$\det(\boldsymbol{A}) = \sum_{i=1}^{N} a_{ij}A_{ij} \quad (\text{第 } j \text{ 列展开}) \tag{20}$$

令 $i=1$,对下面的 2×2 矩阵 A 利用式(19):

$$\boldsymbol{A} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

可得到 $\det\boldsymbol{A} = a_{11}a_{22} - a_{12}a_{21}$。下面的例子显示了如何利用式(19)和式(20)递归地将计算 $N\times N$ 矩阵的行列式简化为计算一系列的 2×2 行列式。

例 3.8 设 $i=1$,利用式(19)计算矩阵 A 的行列式,然后设 $j=2$,利用式(20)计算矩阵 A 的行列式:

$$\boldsymbol{A} = \begin{bmatrix} 2 & 3 & 8 \\ -4 & 5 & -1 \\ 7 & -6 & 9 \end{bmatrix}$$

$i=1$,利用式(19)可得:

$$\begin{aligned} \det\boldsymbol{A} &= (2)\begin{vmatrix} 5 & -1 \\ -6 & 9 \end{vmatrix} - (3)\begin{vmatrix} -4 & -1 \\ 7 & 9 \end{vmatrix} + (8)\begin{vmatrix} -4 & 5 \\ 7 & -6 \end{vmatrix} \\ &= (2)(45-6) - (3)(-36+7) + (8)(24-35) \\ &= 77 \end{aligned}$$

$j=2$,利用式(20)可得:

$$\begin{aligned} \det(\boldsymbol{A}) &= -(3)\begin{vmatrix} -4 & -1 \\ 7 & 9 \end{vmatrix} + (5)\begin{vmatrix} 2 & 8 \\ 7 & 9 \end{vmatrix} - (-6)\begin{vmatrix} 2 & 8 \\ -4 & -1 \end{vmatrix} \\ &= 77 \end{aligned}$$

下面的定理给出了 $\boldsymbol{A}$ 为方阵时,线性方程组 $\boldsymbol{AX}=\boldsymbol{B}$ 的解的存在性和惟一性的充分条件。

定理 3.4 设 $\boldsymbol{A}$ 是 $N\times N$ 方阵,下列命题是等价的:

给定任意 $N\times1$ 矩阵 $\boldsymbol{B}$,线性方程组 $\boldsymbol{AX}=\boldsymbol{B}$ 有惟一解。 (21)

矩阵 $\boldsymbol{A}$ 是非奇异的(即 $\boldsymbol{A}^{-1}$ 存在)。 (22)

方程组 $\boldsymbol{AX}=\boldsymbol{0}$ 有惟一解 $\boldsymbol{X}=\boldsymbol{0}$。 (23)

$\det(\boldsymbol{A})\neq0$ (24)

定理 3.3 和定理 3.4 有助于将矩阵代数与普通代数联系起来。如果命题(21)为真,则命题(22)结合性质(12)和(13)可得出如下推论:

$$\boldsymbol{AX}=\boldsymbol{B} \quad \text{即} \quad \boldsymbol{A}^{-1}\boldsymbol{AX}=\boldsymbol{A}^{-1}\boldsymbol{B}, \quad \text{也即} \quad \boldsymbol{X}=\boldsymbol{A}^{-1}\boldsymbol{B} \tag{25}$$

例 3.9 使用逆矩阵:

$$A^{-1} = \frac{1}{5}\begin{bmatrix} 4 & -1 \\ -7 & 3 \end{bmatrix}$$

和式(25)中的推论,求解线性方程组 $\boldsymbol{AX}=\boldsymbol{B}$:

$$\boldsymbol{AX} = \begin{bmatrix} 3 & 1 \\ 7 & 4 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 2 \\ 5 \end{bmatrix} = \boldsymbol{B}$$

解:

利用式(25)，可得到：

$$X=A^{-1}B=\frac{1}{5}\begin{bmatrix}4 & -1\\ -7 & 3\end{bmatrix}\begin{bmatrix}2\\5\end{bmatrix}=\frac{1}{5}\begin{bmatrix}3\\1\end{bmatrix}=\begin{bmatrix}0.6\\0.2\end{bmatrix}$$

注：在实际计算中，从来不对非奇异矩阵的逆或方阵的行列式进行数值计算。这些概念主要用来建立解的存在性和惟一性的理论“工具”，或作为算术表示线性方程组的解的手段。

3.2.5 平面旋转

设 A 是 3×3 矩阵，$U=[x \quad y \quad z]'$ 是 3×1 矩阵，则乘积 $V=AU$ 是另一个 3×1 矩阵。这是一个线性变换的例子，可在计算机图形学领域中找到相应的应用。矩阵 U 等价于位置向量 $U=(x,y,z)$，表示在三维空间中一个点的位置。考虑下面 3 个特殊矩阵：

$$R_x(\alpha)=\begin{bmatrix}1 & 0 & 0\\ 0 & \cos(\alpha) & -\sin(\alpha)\\ 0 & \sin(\alpha) & \cos(\alpha)\end{bmatrix} \tag{26}$$

$$R_y(\beta)=\begin{bmatrix}\cos(\beta) & 0 & \sin(\beta)\\ 0 & 1 & 0\\ -\sin(\beta) & 0 & \cos(\beta)\end{bmatrix} \tag{27}$$

$$R_z(\gamma)=\begin{bmatrix}\cos(\gamma) & -\sin(\gamma) & 0\\ \sin(\gamma) & \cos(\gamma) & 0\\ 0 & 0 & 1\end{bmatrix} \tag{28}$$

矩阵 $R_x(\alpha)$，$R_y(\beta)$ 和 $R_z(\gamma)$ 分别用来以角度 α,β 和 γ 在 x 轴、y 轴和 z 轴上旋转点。它们的反相为 $R_x(-\alpha)$，$R_y(-\beta)$ 和 $R_z(-\gamma)$，它们在 x 轴、y 轴和 z 轴上旋转的角度分别为 $-\alpha,-\beta$ 和 $-\gamma$。下面的例子描述了这些情况，读者可做进一步研究。

例 3.10　一个单位立方体位于第一象限，一个顶点位于坐标原点。首先在 z 轴上旋转立方体，旋转角度为 $\pi/4$；然后以 $\pi/6$ 在 y 轴上旋转。求旋转后立方体的形状。

解：

第一次旋转的变换矩阵如下所示：

$$V=R_z\left(\frac{\pi}{4}\right)U=\begin{bmatrix}\cos\left(\frac{\pi}{4}\right) & -\sin\left(\frac{\pi}{4}\right) & 0\\ \sin\left(\frac{\pi}{4}\right) & \cos\left(\frac{\pi}{4}\right) & 0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}x\\y\\z\end{bmatrix}$$

$$=\begin{bmatrix}0.707107 & -0.707107 & 0.000000\\ 0.707107 & 0.707107 & 0.000000\\ 0.000000 & 0.000000 & 1.000000\end{bmatrix}\begin{bmatrix}x\\y\\z\end{bmatrix}$$

第二次旋转的变换矩阵如下所示：

$$W=R_y\left(\frac{\pi}{6}\right)V=\begin{bmatrix}\cos\left(\frac{\pi}{6}\right) & 0 & \sin\left(\frac{\pi}{6}\right)\\ 0 & 1 & 0\\ -\sin\left(\frac{\pi}{6}\right) & 0 & \cos\left(\frac{\pi}{6}\right)\end{bmatrix}V$$

$$=\begin{bmatrix} 0.866025 & 0.000000 & 0.500000 \\ 0.000000 & 1.000000 & 0.000000 \\ -0.500000 & 0.000000 & 0.866025 \end{bmatrix} \boldsymbol{V}$$

将二者结合起来,可得:

$$\boldsymbol{W}=\boldsymbol{R}_y\left(\frac{\pi}{6}\right)\boldsymbol{R}_z\left(\frac{\pi}{4}\right)\boldsymbol{U}=\begin{bmatrix} 0.612372 & -0.612372 & 0.500000 \\ 0.707107 & 0.707107 & 0.000000 \\ -0.353553 & 0.353553 & 0.866025 \end{bmatrix}\begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

立方体的顶点坐标的计算结果(表示为位置向量)如表 3.1 所示,变换过程中的立方体形状如图 3.2(a) 到图 3.2(c) 所示。

表 3.1 连续旋转情况下立方体的顶点坐标

$\boldsymbol{U}$	$\boldsymbol{V}=\boldsymbol{R}_z\left(\frac{\pi}{4}\right)\boldsymbol{U}$	$\boldsymbol{W}=\boldsymbol{R}_y\left(\frac{\pi}{6}\right)\boldsymbol{R}_z\left(\frac{\pi}{4}\right)\boldsymbol{U}$
(0,0,0)′	(0.000000,0.000000,0)′	(0.000000,0.000000,0.000000)′
(1,0,0)′	(0.707107,0.707107,0)′	(0.612372,0.707107, -0.353553)′
(0,1,0)′	(-0.707107,0.707107,0)′	(-0.612372,0.707107,0.353553)′
(0,0,1)′	(0.000000,0.000000,1)′	(0.5000000,0.000000,0.866025)′
(1,1,0)′	(0.000000,1.414214,0)′	(0.000000,1.414214,0.000000)′
(1,0,1)′	(0.707107,0.707107,1)′	(1.112372,0.707107,0.512472)′
(0,1,1)′	(-0.707107,0.707107,1)′	(-0.112372,0.707107,1.219579)′
(1,1,1)′	(0.000000,1.414214,1)′	(0.500000,1.414214,0.866025)′

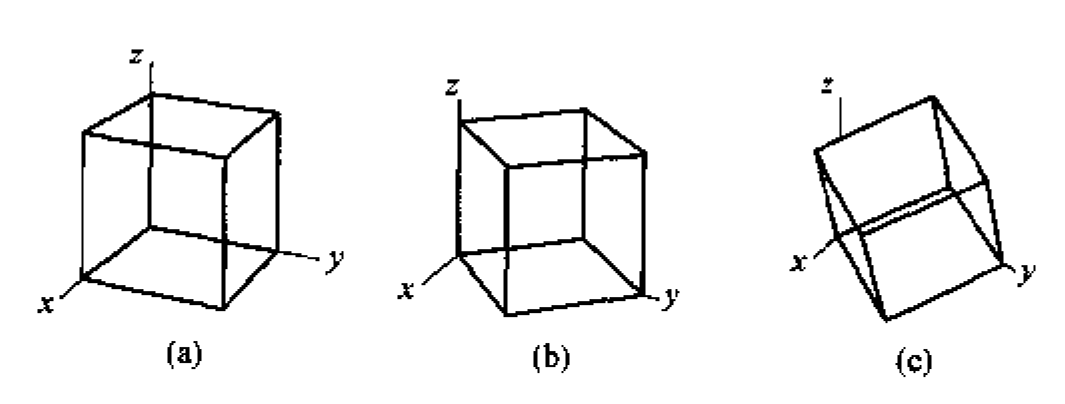


图 3.2 (a) 立方体的初始形状;(b) 在 z 轴进行 $\boldsymbol{V}=\boldsymbol{R}_z(\pi/4)\boldsymbol{U}$ 旋转变换后的形状;(c) 在 y 轴进行 $\boldsymbol{W}=\boldsymbol{R}_y(\pi/6)\boldsymbol{V}$ 旋转变换后的形状

3.2.6 MATLAB

MATLAB 函数 det(A)和 inv(A)分别用来计算方阵 $\boldsymbol{A}$ 的行列式和逆(如果 $\boldsymbol{A}$ 是可逆的)。

例 3.11 使用 MATLAB 和(25)中的逆矩阵法分别求解例 3.6 中的线性方程组。

解:

首先通过证明 $\det(\boldsymbol{A})\neq 0$ (定理 3.4),验证 $\boldsymbol{A}$ 是非奇异矩阵:

```
>>A=[0.125 0.200 0.400;0.375 0.500 0.600;0.500 0.300 0.000];
>>det(A)
ans =
    -0.175
```

然后根据式(25)中的推导,可得到 $AX=B$ 的解是 $X=A^{-1}B$:

```
> >X = inv(A)*[2.3 4.8 2.9]'
X =
      4.0000
      3.0000
      3.0000
```

最后通过检查 $AX=B$ 来验证结果:

```
> >B =A*X
B =
      2.3000
      4.8000
      2.9000
```

3.2.7　向量和矩阵性质的练习

最好手工计算或者使用 MATLAB 进行下面的练习。

1. 针对下列矩阵,求解 AB 和 BA:

$$A=\begin{bmatrix}-3 & 2\\ 1 & 4\end{bmatrix},\quad B=\begin{bmatrix}5 & 0\\ 2 & -6\end{bmatrix}$$

2. 针对下列矩阵,求解 AB 和 BA:

$$A=\begin{bmatrix}1 & -2 & 3\\ 2 & 0 & 5\end{bmatrix},\quad B=\begin{bmatrix}3 & 0\\ -3 & 5\\ 3 & -2\end{bmatrix}$$

3. A,B,C 分别为:

$$A=\begin{bmatrix}3 & 1\\ 0 & 4\end{bmatrix},\quad B=\begin{bmatrix}1 & 2\\ -2 & -6\end{bmatrix},\quad C=\begin{bmatrix}2 & -5\\ 3 & 4\end{bmatrix}$$

(a) 求解 $(AB)C$ 和 $A(BC)$

(b) 求解 $A(B+C)$ 和 $AB+AC$

(c) 求解 $(A+B)C$ 和 $AC+BC$

(d) 求解 $(AB)'$ 和 $B'A'$

4. 设 $A^2=AA$。使用下列矩阵求解 A^2 和 B^2:

$$A=\begin{bmatrix}-1 & -7\\ 5 & 2\end{bmatrix},\quad B=\begin{bmatrix}2 & 0 & 6\\ -1 & 5 & -4\\ 3 & -5 & 2\end{bmatrix}$$

5. 如果下列矩阵的行列式存在,试求之:

(a) $\begin{bmatrix}-1 & -7\\ 5 & 2\end{bmatrix}$　　(b) $\begin{bmatrix}2 & 0 & 6\\ -1 & 5 & -4\\ 3 & -5 & 2\end{bmatrix}$

(c) $\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 0 & 0 \end{bmatrix}$ (d) $\begin{bmatrix} 1 & 2 & 3 & 4 \\ 0 & 2 & 4 & 6 \\ 0 & 0 & 5 & 4 \\ 0 & 0 & 0 & 7 \end{bmatrix}$

6. 使用 $\boldsymbol{R}_x(\alpha)$与 $\boldsymbol{R}_x(-\alpha)$的矩阵乘,证明 $\boldsymbol{R}_x(\alpha)\boldsymbol{R}_x(-\alpha)=\boldsymbol{I}$(参见式(26))。
7. (a) 证明:

$$\boldsymbol{R}_x(\alpha)\boldsymbol{R}_y(\beta)=\begin{bmatrix} \cos(\beta) & 0 & \sin(\beta) \\ \sin(\beta)\sin(\alpha) & \cos(\alpha) & -\cos(\beta)\sin(\alpha) \\ -\cos(\alpha)\sin(\beta) & \sin(\alpha) & \cos(\beta)\cos(\alpha) \end{bmatrix}$$

(参见式(26)和式(27))。

(b) 证明:

$$\boldsymbol{R}_y(\beta)\boldsymbol{R}_x(\alpha)=\begin{bmatrix} \cos(\beta) & \sin(\beta)\sin(\alpha) & \cos(\alpha)\sin(\beta) \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ -\sin(\alpha) & \cos(\beta)\sin(\alpha) & \cos(\beta)\cos(\alpha) \end{bmatrix}$$

8. 如果 $\boldsymbol{A}$ 和 $\boldsymbol{B}$ 为非奇异 $N\times N$ 矩阵,而且 $\boldsymbol{C}=\boldsymbol{AB}$,证明 $\boldsymbol{C}^{-1}=\boldsymbol{B}^{-1}\boldsymbol{A}^{-1}$。提示:利用矩阵乘的结合律。
9. 证明定理 3.3 的命题(13)和命题(16)。
10. 设 $\boldsymbol{A}$ 为 $M\times N$ 矩阵,$\boldsymbol{X}$ 为 $N\times 1$ 矩阵。

(a) 计算 $\boldsymbol{AX}$ 需要多少次乘法?

(b) 计算 $\boldsymbol{AX}$ 需要多少次加法?
11. 设 $\boldsymbol{A}$ 为 $M\times N$ 矩阵,$\boldsymbol{B}$ 和 $\boldsymbol{C}$ 为 $N\times P$ 矩阵。证明矩阵乘的左分配律:$\boldsymbol{A}(\boldsymbol{B}+\boldsymbol{C})=\boldsymbol{AB}+\boldsymbol{AC}$。
12. 设 $\boldsymbol{A}$ 为 $M\times N$ 矩阵,$\boldsymbol{B}$ 和 $\boldsymbol{C}$ 为 $N\times P$ 矩阵。证明矩阵乘的右分配律:$(\boldsymbol{A}+\boldsymbol{B})\boldsymbol{C}=\boldsymbol{AC}+\boldsymbol{BC}$。
13. 设 $\boldsymbol{X}=[1\quad -1\quad 2]$,求解 $\boldsymbol{XX}'$ 和 $\boldsymbol{X}'\boldsymbol{X}$。注意:$\boldsymbol{X}'$ 是 $\boldsymbol{X}$ 的转置。
14. 设 $\boldsymbol{A}$ 为 $M\times N$ 矩阵,$\boldsymbol{B}$ 为 $N\times P$ 矩阵。证明$(\boldsymbol{AB})'=\boldsymbol{B}'\boldsymbol{A}'$。提示:令 $\boldsymbol{C}=\boldsymbol{AB}$,并使用矩阵乘的定义进行证明,即 $\boldsymbol{C}'$ 中的第(i,j)项等于 $\boldsymbol{B}'\boldsymbol{A}'$ 中的第(i,j)项。
15. 利用练习 14 中的结论和矩阵乘的结合律,证明$(\boldsymbol{ABC})'=\boldsymbol{C}'\boldsymbol{B}'\boldsymbol{A}'$。

3.2.8 算法和程序

表 3.1 的第一列包含变换单位立方体的顶点坐标,单位立方体位于第一象限,一个顶点在原点。8 个顶点坐标可用一个 8×3 的矩阵 $\boldsymbol{U}$ 表示,每一行表示一个顶点的坐标。参照练习 14,矩阵 $\boldsymbol{U}$ 与矩阵 $\boldsymbol{R}_z(\pi/4)$的转置相乘可得到一个 8×3 矩阵(如表 3.1 的第二列所示,每一行表示 $\boldsymbol{U}$ 中对应行的变换结果)。结合练习(15)的思想,可认为进行任意次数连续旋转后的立方体顶点坐标可用一个矩阵乘表示。

1. 单位立方体位于第一象限,一个顶点在原点。首先,以角度 $\pi/6$ 沿 y 轴旋转立方体;然后再以角度 $\pi/4$ 沿 z 轴旋转立方体。求旋转后立方体的 8 个顶点坐标,并与例 3.10 的结果进行比较。

它们的区别是什么？试通过矩阵乘一般不满足交换律的解释(参见图 3.3(a) 到(c))。使用 plot3 命令画出三个图形。

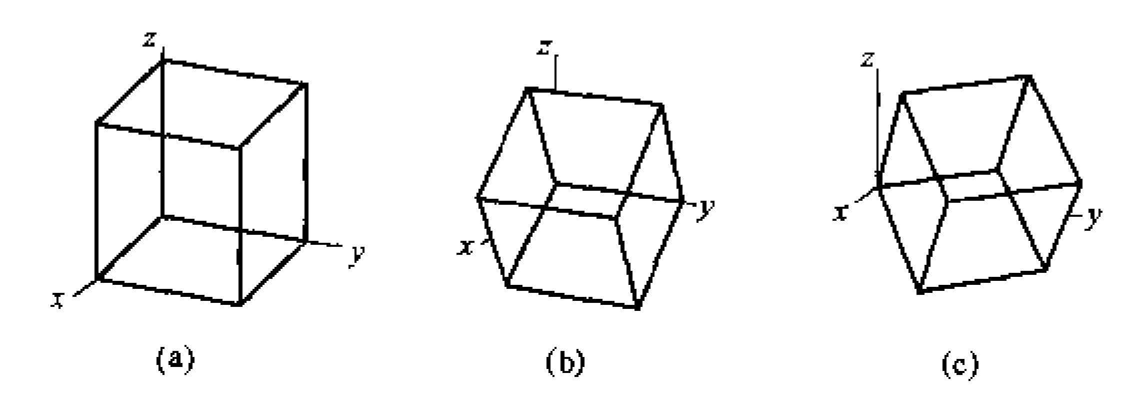


图 3.3　(a) 初始立方体；(b) $V = \boldsymbol{R}_y(\pi/6)\boldsymbol{U}$，沿 y 轴旋转；(c) $W = \boldsymbol{R}_z(\pi/4)$，沿 z 轴旋转

2. 设单位立方体位于第一象限，其中一顶点位于坐标原点。首先以角度 $\pi/12$ 沿 x 轴旋转立方体；然后，再以角度 $\pi/6$ 沿 z 轴旋转立方体。求旋转后立方体 8 个顶点的坐标，并使用 plot3 画出这 3 个立方体。
3. 四面体的坐标为(0,0,0)，(1,0,0)，(0,1,0)，(0,0,1)。首先以弧度 0.15 沿 y 轴旋转，然后再以弧度 -0.15 沿 z 轴旋转，最后以弧度 2.7 沿 x 轴旋转。求旋转后的顶点坐标，并使用 plot3 画出这 4 个立方体。

3.3　上三角线性方程组

现在研究回代算法，它对于由上三角系数矩阵构成的线性方程组的求解很有帮助。这个算法是 3.4 节求解一般线性方程组的算法的一部分。

定义 3.2　$N \times N$ 矩阵 $\boldsymbol{A} = [a_{ij}]$ 中的元素满足对所有，则称 $i > j$，有 $a_{ij} = 0$，则称 $N \times N$，矩阵 $\boldsymbol{A} = [a_{ij}]$ 为上三角矩阵。如果 $\boldsymbol{A}$ 中的元素满足对所有 $i < j$，有 $a_{ij} = 0$，则称矩阵 $\boldsymbol{A}$ 为下三角矩阵。

下面将介绍一种算法来构造上三角线性方程组的解，而将下三角线性方程组的求解留给读者。如果 $\boldsymbol{A}$ 是上三角矩阵，则 $\boldsymbol{AX} = \boldsymbol{B}$ 称为上三角线性方程组，表示为：

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots + a_{1N-1}x_{N-1} + a_{1N}x_N &= b_1 \\ a_{22}x_2 + a_{23}x_3 + \cdots + a_{2N-1}x_{N-1} + a_{2N}x_N &= b_2 \\ a_{33}x_3 + \cdots + a_{3N-1}x_{N-1} + a_{3N}x_N &= b_3 \\ &\vdots \\ a_{N-1N-1}x_{N-1} + a_{N-1N}x_N &= b_{N-1} \\ a_{NN}x_N &= b_N \end{aligned} \tag{1}$$

定理 3.5(回代)　设 $\boldsymbol{AX} = \boldsymbol{B}$ 是(1)中的上三角线方程组。如果：

$$a_{kk} \neq 0, \quad k = 1, 2, \cdots, N \tag{2}$$

则方程组(1)存在惟一解。

证明：用构造法证明。最后一个方程只包含 x_N，因此首先求解这个方程：

$$x_N = \frac{b_N}{a_{NN}} \tag{3}$$

现在 x_N 已知,将它代入上一个方程可得:

$$x_{N-1} = \frac{b_{N-1} - a_{N-1N}x_N}{a_{N-1N-1}} \tag{4}$$

现在可用 x_N 和 x_{N-1} 求解 x_{N-2}:

$$x_{N-2} = \frac{b_{N-2} - a_{N-2N-1}x_{N-1} - a_{N-2}x_N}{a_{N-2N-2}} \tag{5}$$

当 $x_N, x_{N-1}, \cdots, x_{k+1}$ 都求出后,则可得到一般步骤:

$$x_k = \frac{b_k - \sum_{j=k+1}^{N} a_{kj}x_j}{a_{kk}}, \quad k = N-1, N-2, \cdots, 1 \tag{6}$$

可以很容易看出解的惟一性。根据第 N 个方程可推导出 b_N/a_{NN} 是 x_N 的惟一可能值,然后可用有限数学归纳法证明 $x_{N-1}, x_{N-2}, \cdots, x_1$ 是惟一的。

例 3.12 利用回代法求解线性方程组:

$$\begin{aligned} 4x_1 - x_2 + 2x_3 + 3x_4 &= 20 \\ -2x_2 + 7x_3 - 4x_4 &= -7 \\ 6x_3 + 5x_4 &= 4 \\ 3x_4 &= 6 \end{aligned}$$

求解最后一个方程中的 x_4 可得:

$$x_4 = \frac{6}{3} = 2$$

将 $x_4 = 2$ 代入第三个方程,可得:

$$x_3 = \frac{4-5(2)}{6} = 1$$

现在将 $x_3 = -1$ 和 $x_4 = 2$ 代入第二个方程,可得:

$$x_2 = \frac{-7-7(-1)+4(2)}{-2} = -4$$

最后,求解第一个方程中的 x_1 可得:

$$x_1 = \frac{20+1(-4)-2(-1)-3(2)}{4} = 3$$

条件 $a_{kk} \neq 0$ 非常重要,因为式(6)包含对 a_{kk} 的除法。如果条件不满足,则可能无解或有无穷解。

例 3.13 证明下列线性方程组无解:

$$\begin{aligned} 4x_1 - x_2 + 2x_3 + 3x_4 &= 20 \\ 0x_2 + 7x_3 - 4x_4 &= -7 \\ 6x_3 + 5x_4 &= 4 \\ 3x_4 &= 6 \end{aligned} \tag{7}$$

证明:

求解式(7)中的最后一个方程可得 $x_4 = 2$,将它代入第二个方程和第三个方程可得:

$$7x_3 - 8 = -7$$

$$6x_3+10=4 \tag{8}$$

求解式(8)中的第一个方程可得 $x_3=1/7$，而求解第二个方程可得 $x_3=-1$。二者矛盾，所以线性方程组(7)无解。

例 3.14　证明下列线性方程组有无穷解：

$$\begin{aligned} 4x_1 - x_2 + 2x_3 + 3x_4 &= 20 \\ 0x_2 + 7x_3 + 0x_4 &= -7 \\ 6x_3 + 5x_4 &= 4 \\ 3x_4 &= 6 \end{aligned} \tag{9}$$

证明：

利用式(9)中的最后一个方程，可得到 $x_4=2$，将它代入第二个方程和第三个方程可得 $x_3=-1$，但对第二个方程到第四个方程的求解后，只能得到两个解 x_3 和 x_4。当将它们代入第一个方程，可得：

$$x_2=4x_1-16 \tag{10}$$

而式(10)有无穷解，因此式(9)也有无穷解。如果对式(10)中的 x_1 选定一个值，则 x_2 的值是惟一的。例如在式(9)中增加一个方程 $x_1=2$，则可计算出 $x_2=-8$。

定理 3.4 指出若 $\boldsymbol{A}$ 为 $N\times N$ 矩阵，线性方程组 $\boldsymbol{AX}=\boldsymbol{B}$ 有惟一解当且仅当 $\det(\boldsymbol{A})\neq0$。下面的定理指出如果上三角矩阵或下三角矩阵中主对角线的任一元素为零，则 $\det(\boldsymbol{A})=0$。这样，观察前面三个线性方程组，可清楚地发现例 3.12 有惟一解，而例 3.13 和例 3.14 没有惟一解。定理 3.6 的证明可在大多数线性代数教材中找到。

定理 3.6　如果 $N\times N$ 矩阵 $\boldsymbol{A}=[a_{ij}]$ 是上三角矩阵或下三角矩阵，则：

$$\det(\boldsymbol{A}) = a_{11}a_{22}\cdots a_{NN} = \prod_{i=1}^{N} a_{ii} \tag{11}$$

例 3.12 中系数矩阵的行列式值为 $\det(\boldsymbol{A})=4(-2)(6)(3)=-144$。例 3.13 和例 3.14 中系数矩阵的行列式值都为 $4(0)(6)(3)=0$。

下面的程序利用回代法求解上三角线性方程组(1)的解，设 $a_{kk}\neq0$，其中 $k=1,2,\cdots,N$。

程序 3.1(回代)　用回代法求解上三角线性方程组 $\boldsymbol{AX}=\boldsymbol{B}$，必须满足系数矩阵的对角元素非零的条件。首先计算 $x_N=b_N/a_{NN}$，然后利用如下表达式：

$$x_k = \frac{b_k \sum_{j=k+1}^{N} a_{kj}x_j}{a_{kk}}, k=N-1,N-2,\cdots,1$$

```
function X = backsub(A,B)
% Input  - A is an n x n upper-triangular nonsingular matrix
%        - B is an n x 1 matrix
% Output - X is the solution to the linear system AX = B
% Find the dimension of B and initialize X
n = length(B);
X = zeros(n,1);
```

```
X(n) = B(n)/A(n,n);
for k = n-1: -1:1
   x(k) = (B(k) - A(k,k+1:n) * X(K+1:n))/A(k,k);
end
```

3.3.1 上三角线性方程组的练习

求解练习 1 到练习 3 中的上三角线性方程组,并求解系数矩阵的行列式值。

1.
$$\begin{aligned} 3x_1 - 2x_2 + x_3 - x_4 &= 8 \\ 4x_2 - x_3 + 2x_4 &= -3 \\ 2x_3 + 3x_4 &= 11 \\ 5x_4 &= 15 \end{aligned}$$

2.
$$\begin{aligned} 5x_1 - 3x_2 - 7x_3 + x_4 &= -14 \\ 11x_2 + 9x_3 + 5x_4 &= 22 \\ 3x_3 - 13x_4 &= -11 \\ 7x_4 &= 14 \end{aligned}$$

3.
$$\begin{aligned} 4x_1 - x_2 + 2x_3 + 2x_4 - x_5 &= 4 \\ -2x_2 + 6x_3 + 2x_4 + 7x_5 &= 0 \\ x_3 - x_4 - 2x_5 &= 3 \\ -2x_4 - x_5 &= 10 \\ 3x_5 &= 6 \end{aligned}$$

4. (a) 设有两个上三角矩阵:

$$\boldsymbol{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a_{22} & a_{23} \\ 0 & 0 & a_{33} \end{bmatrix} \quad 和 \quad \boldsymbol{B} = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ 0 & b_{22} & b_{23} \\ 0 & 0 & b_{33} \end{bmatrix}$$

证明它们的乘积 $\boldsymbol{C} = \boldsymbol{AB}$ 是上三角矩阵。

(b) 设 $\boldsymbol{A}$ 和 $\boldsymbol{B}$ 为两个 $N \times N$ 上三角矩阵。证明它们的乘积为上三角矩阵。

5. 求解下三角线性方程组 $\boldsymbol{AX} = \boldsymbol{B}$,并求解 $\det(\boldsymbol{A})$:

$$\begin{aligned} 2x_1 \qquad\qquad &= 6 \\ -x_1 + 4x_2 \qquad &= 5 \\ 3x_1 - 2x_2 - x_3 \qquad &= 4 \\ x_1 - 2x_2 + 6x_3 + 3x_4 &= 2 \end{aligned}$$

6. 求解下三角线性方程组 $\boldsymbol{AX} = \boldsymbol{B}$,并求解 $\det(\boldsymbol{A})$:

$$\begin{aligned} 5x_1 \qquad\qquad &= -10 \\ x_1 + 3x_2 \qquad &= 4 \\ 3x_1 - 4x_2 - 2x_3 \qquad &= 2 \\ -x_1 - 3x_2 + 6x_3 + x_4 &= 5 \end{aligned}$$

7. 证明回代法需要 N 次除法、$(N^2 - N)/2$ 次乘法和 $(N^2 - N)/2$ 次加法或减法。提示:可利用下列公式:

$$\sum_{k=1}^{M} k = M(M+1)/2$$

3.3.2 算法和程序

1. 使用程序 3.1 求解方程组 $\boldsymbol{UX} = \boldsymbol{B}$,表示如下:

$$U=[u_{ij}]_{10\times 10}\quad,\quad u_{ij}=\begin{cases}\cos(ij) & i\leqslant j\\ 0 & i>j\end{cases}$$

而且 $\boldsymbol{B}=[b_{i1}]_{10\times 1}$，$b_{i1}=\tan(i)$。

2. 前向替换算法。对于线性方程组 $\boldsymbol{AX}=\boldsymbol{B}$，如果满足当 $i<j$ 时，$a_{ij}=0$，则称其为下三角线性方程组。构造类似程序 3.1 的程序 forsub，用其求解下列下三角线性方程组。注：此程序将在 3.5 节中使用。

$$\begin{aligned}
&a_{11}x_1 &&= b_1\\
&a_{21}x_1+\quad a_{22}x_2 &&= b_2\\
&a_{31}x_1+\quad a_{32}x_2+\quad a_{33}x_3 &&= b_3\\
&\vdots\qquad\quad \vdots\qquad\quad \vdots &&\ \ \vdots\\
&a_{N-1\,1}x_1+a_{N-1\,2}x_2+a_{N-1\,3}x_3+\cdots+a_{N-1\,N-1}x_{N-1} &&= b_{N-1}\\
&a_{N1}x_1+\quad a_{N2}x_2+\quad a_{N3}x_3+\cdots+\quad a_{NN-1}x_{N-1}+a_{NN}x_N &&= b_N
\end{aligned}$$

3. 使用 forsub 求解方程组 $\boldsymbol{LX}=\boldsymbol{B}$，其中：

$$\boldsymbol{L}=[l_{ij}]_{20\times 20},\ l_{ij}=\begin{cases}i+j & i\geqslant j\\ 0 & i<j\end{cases}\quad \text{而且 } \boldsymbol{B}=[b_{i1}]_{20\times 1},\ b_{i1}=i$$

3.4 高斯消去法和选主元

在这一节里，将研究求解有 N 个方程和 N 个未知数的一般方程组 $\boldsymbol{AX}=\boldsymbol{B}$。目标是构造一个等价的上三角方程组 $\boldsymbol{UX}=\boldsymbol{Y}$，这样可以利用 3.3 节中的方法进行求解。

如果两个 $N\times N$ 线性方程组的解相同，则称二者等价。根据线性代数中的定理可知，对一个给定方程组进行一定的变换，不会改变它的解。

定理 3.7（初等变换） 下面三种变换可使一个线性方程组变换成另一个等价的线性方程组：

交换变换：对调方程组的两行　(1)

比例变换：用非零常数乘方程组的某一行　(2)

替换变换：将方程组的某一行乘一个常数，再加到另一行上去　(3)

通常利用式(3)，即用一个方程乘一个常数，再减去另一个方程来替换另一个方程。请参看下面的例子以了解这些概念。

例 3.15 求抛物线 $y=A+Bx+Cx^2$，它经过三点(1,1)，(2,−1)，(3,1)。

对每个点，可得到一个方程，并形成线性方程组：

$$\begin{aligned}
A+\ B+\ C&=\ \ 1 &&\text{在点}(1,1)\\
A+2B+4C&=-1 &&\text{在点}(2,-1)\\
A+3B+9C&=\ \ 1 &&\text{在点}(3,1)
\end{aligned}\tag{4}$$

使用第二个方程和第三个方程减去第一个方程可消去变量 A。这是利用替换变换式(3)。等价的线性方程组如下：

$$\begin{aligned}
A+B+\ C&=\ \ 1\\
B+3C&=-2\\
2B+8C&=\ \ 0
\end{aligned}\tag{5}$$

使用(5)中的第三个方程减去第二个方程的两倍可消去变量 B。等价的上三角线性方程组为:

$$\begin{aligned} A+B+\ C &= \ \ 1 \\ B+3C &= -2 \\ 2C &= \ \ 4 \end{aligned} \tag{6}$$

现在可以利用回代法求出系数 $C=4/2=2$, $B=-2-3(2)=-8$ 和 $A=1-(-8)-2=7$,这样抛物线方程为 $y=7-8x+2x^2$。

可以将线性方程组 $\boldsymbol{AX}=\boldsymbol{B}$ 的系数保存在一个 $N\times(N+1)$的数组中。$\boldsymbol{B}$ 的系数保存在这个数组的 $N+1$ 列中(即 $a_{kN+1}=b_k$)。这样每一行包含线性方程组中的每个方程的系数。增广矩阵表示为$[\boldsymbol{A}|\boldsymbol{B}]$,线性方程组可表示为:

$$[\boldsymbol{A}|\boldsymbol{B}]=\left[\begin{array}{cccc|c} a_{11} & a_{12} & \cdots & a_{1N} & b_1 \\ a_{21} & a_{22} & \cdots & a_{2N} & b_2 \\ \vdots & \vdots & & \vdots & \vdots \\ a_{N1} & a_{N2} & \cdots & a_{NN} & b_N \end{array}\right] \tag{7}$$

方程组 $\boldsymbol{AX}=\boldsymbol{B}$ 的增广矩阵如(7)所示,可通过对增广矩阵$[\boldsymbol{A}|\boldsymbol{B}]$进行行变换求解方程。变量 x_k 是系数的占位符,在计算结束之前可以忽略。

定理 3.8 (初等行变换) 对增广矩阵式(7)进行如下的变换可得到一个等价的线性方程组:

交换行变换:对调矩阵的两行 (8)

比例行变换:用非零常数乘矩阵某一行的所有元素 (9)

替换行变换:将矩阵的某一行的所有元素乘一个常数,再加到另一行对应的元素上去,即:

$$\text{row}_r=\text{row}_r-m_{rp}\times\text{row}_p \tag{10}$$

通常可利用式(10)将矩阵的一行乘一个常数,再减去另一行来替换另一行。

定义 3.3 (主元) 系数矩阵 $\boldsymbol{A}$ 中的元素 a_{rr}用来消去 a_{kr},其中 $k=r+1,r+2,\cdots,N$,这里称 a_{rr}为第 r 个主元(***pivotal element***),第 r 行称为主元行(***pivot row***)。

下列例子显示了如何利用定理 3.8 中的变换,从线性方程组 $\boldsymbol{AX}=\boldsymbol{B}$ 得到一个等价的上三角线性方程组 $\boldsymbol{UX}=\boldsymbol{Y}$,这里 $\boldsymbol{A}$ 为$N\times N$ 矩阵。

例 3.16 用增广矩阵表示下列线性方程组,并求等价的上三角线性方程组和方程组的解:

$$\begin{aligned} x_1+2x_2+\ x_3+4x_4 &= 13 \\ 2x_1+0x_2+4x_3+3x_4 &= 28 \\ 4x_1+2x_2+2x_3+\ x_4 &= 20 \\ -3x_1+x_2+3x_3+2x_4 &= \ 6 \end{aligned}$$

解:

增广矩阵为:

$$\begin{array}{l} \text{主元}\rightarrow \\ m_{21}=2 \\ m_{31}=4 \\ m_{41}=-3 \end{array}\left[\begin{array}{cccc|c} \underline{1} & 2 & 1 & 4 & 13 \\ 2 & 0 & 4 & 3 & 28 \\ 4 & 2 & 2 & 1 & 20 \\ -3 & 1 & 3 & 2 & 6 \end{array}\right]$$

用行 1 消去列 1 中对角线下的元素。将行 1 作为主元行，元素 $a_{11}=1$ 作为主元。用行 1 乘常数 m_{k1}，再被行 k 减，$k=2,3,4$。结果如下：

$$\begin{array}{r} \\ \text{主元}\rightarrow \\ m_{32}=1.5 \\ m_{42}=-1.75 \end{array}\left[\begin{array}{rrrr|r} 1 & 2 & 1 & 4 & 13 \\ 0 & \underline{-4} & 2 & -5 & 2 \\ 0 & -6 & -2 & -15 & -32 \\ 0 & 7 & 6 & 14 & 45 \end{array}\right]$$

用行 2 消去列 2 中对角线下的元素。将行 2 作为主元行，用行 2 乘常数 m_{k2}，再被行 k 减，$k=3,4$。结果如下：

$$\begin{array}{r} \\ \\ \text{主元}\rightarrow \\ m_{43}=1.9 \end{array}\left[\begin{array}{rrrr|r} 1 & 2 & 1 & 4 & 13 \\ 0 & -4 & 2 & -5 & 2 \\ 0 & 0 & \underline{-5} & -7.5 & -35 \\ 0 & 0 & 9.5 & 5.25 & 48.5 \end{array}\right]$$

最后用行 3 乘常数 $m_{k3}=-1.9$，再被行 4 减，结果是上三角线性方程组的增广矩阵，表示如下：

$$\left[\begin{array}{rrrr|r} 1 & 2 & 1 & 4 & 13 \\ 0 & -4 & 2 & -5 & 2 \\ 0 & 0 & -5 & -7.5 & -35 \\ 0 & 0 & 0 & -9 & -18 \end{array}\right] \tag{11}$$

用回代法求解式(11)，可得到：

$$x_4=2,\quad x_3=4,\quad x_2=-1,\quad x_1=3$$

上述过程称为高斯消去法，但必须对其进行改进以适用于大多数情况。如果 $a_{kk}=0$，则不能使用第 k 行消除第 k 列的元素，而需要将第 k 行与对角线下的某行进行交换，以得到一个非零主元。如果不能找到非零主元，则线性方程组的系数矩阵是奇异的，因此线性方程组不存在惟一解。

定理 3.9（有回代的高斯消去法）　如果 $\boldsymbol{A}$ 是 $N\times N$ 非奇异矩阵，则存在线性方程组 $\boldsymbol{UX}=\boldsymbol{Y}$ 与线性方程组 $\boldsymbol{AX}=\boldsymbol{B}$ 等价，这里 $\boldsymbol{U}$ 是上三角矩阵，并且 $u_{kk}\neq 0$。当构造出 $\boldsymbol{U}$ 和 $\boldsymbol{Y}$ 后，可用回代法求解 $\boldsymbol{UX}=\boldsymbol{Y}$，并得到方程组的解 $\boldsymbol{X}$。

证明：首先将使用带 $N+1$ 列矩阵 $\boldsymbol{B}$ 的增广矩阵：

$$\boldsymbol{AX}=\begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} & \cdots & a_{1N}^{(1)} \\ a_{21}^{(1)} & a_{22}^{(1)} & a_{23}^{(1)} & \cdots & a_{2N}^{(1)} \\ a_{31}^{(1)} & a_{32}^{(1)} & a_{33}^{(1)} & \cdots & a_{3N}^{(1)} \\ \vdots & \vdots & \vdots & & \vdots \\ a_{N1}^{(1)} & a_{N2}^{(1)} & a_{N3}^{(1)} & \cdots & a_{NN}^{(1)} \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_N \end{bmatrix}=\begin{bmatrix} a_{1\,N+1}^{(1)} \\ a_{2\,N+1}^{(1)} \\ a_{3\,N+1}^{(1)} \\ \vdots \\ a_{N\,N+1}^{(1)} \end{bmatrix}=\boldsymbol{B}$$

然后将构造等价的上三角线性方程组 $\boldsymbol{UX}=\boldsymbol{Y}$：

$$UX=\begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} & \cdots & a_{1N}^{(1)} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & \cdots & a_{2N}^{(2)} \\ 0 & 0 & a_{33}^{(3)} & \cdots & a_{3N}^{(3)} \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \cdots & a_{NN}^{(N)} \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_N \end{bmatrix}=\begin{bmatrix} a_{1\,N+1}^{(1)} \\ a_{2\,N+1}^{(2)} \\ a_{3\,N+1}^{(3)} \\ \vdots \\ a_{N\,N+1}^{(N)} \end{bmatrix}=Y$$

第一步:将系数保存在增广矩阵中。$a_{rc}^{(1)}$的上标表示第一次保存在位置(r,c)中的元素:

$$\left[\begin{array}{ccccc|c} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} & \cdots & a_{1N}^{(1)} & a_{1\,N+1}^{(1)} \\ a_{21}^{(1)} & a_{22}^{(1)} & a_{23}^{(1)} & \cdots & a_{2N}^{(1)} & a_{2\,N+1}^{(1)} \\ a_{31}^{(1)} & a_{32}^{(1)} & a_{33}^{(1)} & \cdots & a_{3N}^{(1)} & a_{3\,N+1}^{(1)} \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ a_{N1}^{(1)} & a_{N2}^{(1)} & a_{N3}^{(1)} & \cdots & a_{NN}^{(1)} & a_{N\,N+1}^{(1)} \end{array}\right]$$

第二步:如果有必要,交换行使得 $a_{11}^{(1)}\neq 0$;然后消去从行 2 到行 N 的 x_1。在这个过程中,m_{r1}是行 r 减去行 1 后,行 1 的倍数:

```
for r = 2 : N
  m_r1 = a_r1^(1)/a_11^(1);
  a_r1^(2) = 0;
  for c = 2 : N+1
    a_rc^(2) = a_rc^(1) - m_r1 * a_1c^(1);
  end
end
```

新的元素表示为 $a_{rc}^{(2)}$,它表示第二次保存在矩阵中,位置为(r,c)的元素。执行第二步后的结果为:

$$\left[\begin{array}{ccccc|c} a_{11}^{(1)} & a_{12}^{(1)} & a_{(13)}^{(1)} & \cdots & a_{1N}^{(1)} & a_{1\,N+1}^{(1)} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & \cdots & a_{2N}^{(2)} & a_{2\,N+1}^{(2)} \\ 0 & a_{32}^{(2)} & a_{33}^{(2)} & \cdots & a_{3N}^{(2)} & a_{3\,N+1}^{(2)} \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & a_{N2}^{(2)} & a_{N3}^{(2)} & \cdots & a_{NN}^{(2)} & a_{N\,N+1}^{(2)} \end{array}\right]$$

第三步:如果有必要,将行 2 与它下面的某行进行交换,使得 $a_{22}^{(2)}\neq 0$;然后消去行 3 到行 N 中的 x_2。在这个过程中,m_{r2}是行 r 减去行 2 后,行 2 的倍数:

```
for r = 3 : N
  m_r2 = a_r2^(2)/a_22^(2);
  a_r2^(3) = 0;
  for c = 3:N+1
    a_rc^(3) = a_rc^(2) - m_r2 * a_2c^(2);
  end
end
```

新的元素表示为 $a_{rc}^{(3)}$，它表示第三次保存在矩阵中，位置为 (r,c) 的元素。执行第三步后的结果为：

$$\left[\begin{array}{ccccc|c} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} & \cdots & a_{1N}^{(1)} & a_{1\,N+1}^{(1)} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & \cdots & a_{2N}^{(2)} & a_{2\,N+1}^{(2)} \\ 0 & 0 & a_{33}^{(3)} & \cdots & a_{3N}^{(3)} & a_{3\,N+1}^{(3)} \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & a_{N3}^{(3)} & \cdots & a_{NN}^{(3)} & a_{N\,N+1}^{(3)} \end{array}\right]$$

第 $p+1$ 步：这是一般步骤。如果有必要，将第 p 行与它下面的某行进行交换，使得 $a_{pp}^{(p)}\neq 0$，然后消去行 $p+1$ 到行 N 中的 x_p。在这个过程中，m_{rp} 是行 r 减去行 p 后，行 p 的倍数：

```
for r = p+1:N
    m_rp = a_rp^(p)/a_pp^(p);
    a_rp^(p+1) = 0;
    for c = p+1:N + 1
        a_rc^(p+1) = a_rc^(p) - m_rp * a_pc^(p);
    end
end
```

当行 N 中的 x_{N-1} 被消去后，结果为：

$$\left[\begin{array}{ccccc|c} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} & \cdots & a_{1N}^{(1)} & a_{1\,N+1}^{(1)} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & \cdots & a_{2N}^{(2)} & a_{2\,N+1}^{(2)} \\ 0 & 0 & a_{33}^{(3)} & \cdots & a_{3N}^{(3)} & a_{3\,N+1}^{(3)} \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & a_{NN}^{(N)} & a_{N\,N+1}^{(N)} \end{array}\right]$$

上三角矩阵构造过程执行完毕。

由于 $\boldsymbol{A}$ 是非奇异的，所以当执行完行变换后的矩阵仍是非奇异的。这保证了在构造过程中，对所有的 k，有 $a_{kk}^{(k)}\neq 0$。因此可利用回代法求 $\boldsymbol{UX}=\boldsymbol{Y}$ 的解 $\boldsymbol{X}$。定理得证。

3.4.1 选主元以避免 $a_{pp}^{(p)}=0$

如果 $a_{pp}^{(p)}=0$，则不能使用第 p 行消去主对角线之下列 p 的元素。有必要寻找行 k，满足 $a_{kp}^{(p)}\neq 0$ 而且 $k>p$，然后交换行 k 和行 p，使得主元非零，这个过程称为选主元，选择行的判定条件称为选主元策略。平凡选主元策略(trivial pivoting strategy)是：如果 $a_{pp}^{(p)}\neq 0$，不交换行；如果 $a_{pp}^{(p)}=0$，寻找第 p 行下满足 $a_{kp}^{(p)}\neq 0$ 的第一行，设行数为 k，然后交换行 k 和行 p。这导致新元素 $a_{pp}^{(p)}\neq 0$ 是非零主元。

3.4.2 选主元以减少误差

由于计算机使用固定精度计算，这样在每次算术计算中可能引入微小的误差。下面的例子显示在采用高斯消去法求解线性方程组中，使用平凡选主元策略如何导致巨大的误差。

例 3.17 值 $x_1 = x_2 = 1.000$ 是如下方程组的解：

$$\begin{aligned}1.133x_1 + 5.281x_2 &= 6.414\\ 24.14x_1 - 1.210x_2 &= 22.93\end{aligned} \tag{12}$$

使用 4 位有效数字精度计算(参见 1.3 节的练习 6 和练习 7)以及采用平凡选主元策略的高斯消去法，求解上述线性方程组解的近似值。

行 2 减去行 1 乘倍数 $m_{21} = 24.14/1.133 = 21.31$ 得到上三角线性方程组。使用 4 位有效数字精度计算，可得到新的系数：

$$a_{22}^{(2)} = -1.210 - 21.31(5.281) = -1.210 - 112.5 = -113.7$$
$$a_{23}^{(2)} = 22.93 - 21.31(6.414) = 22.93 - 136.7 = -113.8$$

计算后的上三角线性方程组为：

$$\begin{aligned}1.133x_1 + 5.281x_2 &= 6.414\\ -113.7x_2 &= -113.8\end{aligned}$$

利用回代法可得到 $x_2 = -113.8/(-113.7) = 1.001$ 和 $x_1 = (6.414 - 5.281(1.001))/(1.133) = (6.414 - 5.286)/(1.133) = 0.9956$。

线性方程组(12)解的误差是由于倍数 $m_{21} = 21.31$ 的值引起的。在下述例子中，通过交换线性方程组(12)中第 1 行和第 2 行来减少倍数的值，然后利用平凡选主元策略的高斯消去法求解线性方程组。

例 3.18 使用 4 位有效数字精度计算和平凡选主元策略的高斯消去法，求解线性方程组：

$$\begin{aligned}24.14x_1 - 1.210x_2 &= 22.93\\ 1.133x_1 + 5.281x_2 &= 6.414\end{aligned}$$

这次用行 2 减去行 1 乘倍数 $m_{21} = 1.133/24.14 = 0.04693$。新的系数为：

$$a_{22}^{(2)} = 5.281 - 0.04693(-1.210) = 5.281 + 0.05679 = 5.338$$
$$a_{23}^{(2)} = 6.414 - 0.04693(22.93) = 6.414 - 1.076 = 5.338$$

计算后的上三角线性方程组为：

$$\begin{aligned}24.14x_1 - 1.210x_2 &= 22.93\\ 5.338x_2 &= 5.338\end{aligned}$$

利用回代法可得到 $x_2 = 5.338/5.338 = 1.000$ 和 $x_1 = (22.93 + 1.210(1.000))/(24.14) = 1.000$。

选主元策略的目的在于将元素中的最大绝对值移到主对角线上，然后用其消去列中的剩余元素。如果在列 p 中存在多个非零元素，则要从中选择一个进行行交换。例 3.18 中的偏序选主元策略(partial pivoting strategy)是最常用的一个，而且在程序 3.2 中使用。为了减少误差的传播，偏序选主元策略首先检查位于主对角线或主对角线下列 p 的所有元素，确定行 k，它的元素的绝对值最大，即：

$$|a_{kp}| = \max\{|a_{pp}|, |a_{p+1\,p}|, \cdots, |a_{N-1\,p}|, |a_{Np}|\}$$

然后如果 $k > p$，则交换行 k 和行 p。现在，每个倍数 m_{rp} 的绝对值，将小于或等于 1，$r = p + 1, \cdots, N$。这样保证了定理 3.9 中的矩阵 U 与初始系数矩阵 A 的对应元素的相对大小一致。在

偏序选主元策略中，通常选择更大的主元元素会导致更小的传播误差。

在 3.5 节中，可以看到求解 $N \times N$ 线性方程组总共需要 $(4N^3 + 9N^2 - 7N)/6$ 次算术操作。当 $N = 20$，则总的算术操作次数为 5910，在计算过程中的误差传播将导致错误的结果。按比例偏序选主元(scaled partial pivoting)策略或平衡(equilibrating)策略可用来进一步减少误差传播。在按比例偏序选主元法中，搜索位于主对角线或主对角线下列 p 的元素，此元素满足在它所在行中，它的绝对值相对最大。首先搜索行 p 到行 N 中的绝对值最大的元素，称为 s_r：

$$s_r = \max\{|a_{rp}|, |a_{rp+1}|, \cdots, |a_{rN}|\}, \quad r = p, p+1, \cdots, N \tag{13}$$

通过求解下式确定行 k：

$$\frac{|a_{kp}|}{s_k} = \max\left\{\frac{|a_{pp}|}{s_p}, \frac{|a_{p+1p}|}{s_{p+1}}, \cdots, \frac{a_{Np}}{s_N}\right\} \tag{14}$$

现在交换行 p 和行 k，除非 $p = k$。这样也是为了保证定理 3.9 中的矩阵 $\boldsymbol{U}$ 与初始系数矩阵 $\boldsymbol{A}$ 的对应元素的相对大小一致。

3.4.3 病态情况

如果存在矩阵 $\boldsymbol{B}$，当矩阵 $\boldsymbol{B}$ 和矩阵 $\boldsymbol{A}$ 中系数元素的微小变化使得 $\boldsymbol{X} = \boldsymbol{A}^{-1}\boldsymbol{B}$ 变化很大，则称矩阵 $\boldsymbol{A}$ 为病态矩阵。如果矩阵 $\boldsymbol{A}$ 为病态矩阵，则方程组 $\boldsymbol{AX} = \boldsymbol{B}$ 称为病态方程组。在这种情况下，计算解的近似值的数值方法将产生很大的误差。

当 $\boldsymbol{A}$ 近似于奇异而且它的行列式接近 0 时，可能发生病态情况。当两个方程表示的直线接近平行(或三个方程表示的三个平面接近平行)时，它们组成的方程组也可能是病态的。发生病态情况可能导致错误解。例如，设有下面两个方程：

$$\begin{aligned} x + 2y - 2.00 &= 0 \\ 2x + 3y - 3.40 &= 0 \end{aligned} \tag{15}$$

将 $x_0 = 1.00$ 和 $y_0 = 0.48$ 代入这些"几乎等于零"的方程可得到：

$$\begin{aligned} 1 + 2(0.48) - 2.00 &= 1.96 - 2.00 = -0.04 \approx 0 \\ 2 + 3(0.48) - 3.40 &= 3.44 - 3.40 = 0.04 \approx 0 \end{aligned}$$

这里结果与 0 的偏差只有 ± 0.04。而线性方程组解的真实值为 $x = 0.8$ 和 $y = 0.6$，因此近似值解的误差为 $x - x_0 = 0.80 - 1.00 = -0.20$ 和 $y - y_0 = 0.60 - 0.48 = 0.12$。所以仅仅将值代入方程组中不是很可靠的测试方法。图 3.4 中的菱形区间 R 表示"几乎满足"式(15)中的方程的近似值集合：

$$R = \{(x, y): |x + 2y - 2.00| < 0.1 \quad 且 \quad |2x + 3y - 3.40| < 0.2\}$$

在区间 R 中某些点远离解(0.8,0.6)，但代入式(15)中的方程后，得到的值很小。如果怀疑一个线性方程组是病态的，则应该用多精度算术计算。有兴趣的读者可研究有关矩阵的条件数的主题，可得到这方面的更多信息。

当包含多个方程时，病态情况可能使结果变化很大。设求解三次多项式 $y = c_1x^3 + c_2x^2 + c_3x + c_4$，它经过四个点(2,8)，(3,27)，(4,64)，(5,125)(很明显，$y = x^3$ 是要找的三次多项式)。在第 5 章，将介绍最小二乘法。利用最小二乘法寻找系数时需要求解如下线性方程组：

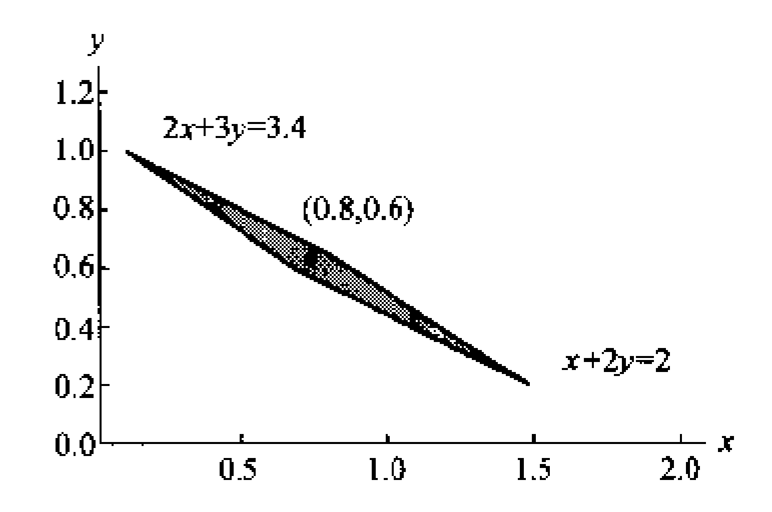


图 3.4 两个方程都“几乎满足”的区域

$$\begin{bmatrix} 20\,514 & 4\,424 & 978 & 224 \\ 4\,424 & 978 & 224 & 54 \\ 978 & 224 & 54 & 14 \\ 224 & 54 & 14 & 4 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{bmatrix} = \begin{bmatrix} 20\,514 \\ 4\,424 \\ 978 \\ 224 \end{bmatrix}$$

使用一个精度有 9 位有效数字的计算机计算系数可得:

$$c_1 = 1.000004,\quad c_2 = -0.000038,\quad c_3 = 0.000126,\quad c_4 = -0.000131$$

尽管计算结果接近真实值 $c_1 = 1, c_2 = c_3 = c_4 = 0$,但可看出结果中很容易产生误差。如果进一步将上三角矩阵中的系数 $a_{11} = 20\,514$ 改为 $a_{11} = 20\,515$,并求解这个被扰动的方程组。使用同样的计算机,计算的结果为:

$$c_1 = 0.642857,\quad c_2 = 3.75000,\quad c_3 = -12.3928 \text{ 和 } c_4 = 12.7500$$

这个答案的意义不大,病态情况不易被检查出。如果轻微扰动方程组的系数,使得结果改变很大,则可认为线性方程组为病态线性方程组。通常在高级数值分析教材中详细介绍了与此相关的灵敏度分析。

3.4.4 MATLAB

在程序 3.2 中,MATLAB 语句[A B]用来构造线性方程组 $\boldsymbol{AX} = \boldsymbol{B}$ 的增广矩阵,max 命令用于偏序选主元策略中的主元选择。一旦得到等价的上三角矩阵$[\boldsymbol{U}|\boldsymbol{Y}]$,将它分成 $\boldsymbol{U}$ 和 $\boldsymbol{Y}$,程序 3.1 用来执行回代法(backsub(U,Y))。在下面的例子中显示了对这些过程和命令的使用情况。

例 3.19 (a) 使用 MATLAB 构造例 3.16 中的增广矩阵;(b) 使用 max 命令求系数矩阵 $\boldsymbol{A}$ 列 1 中绝对值最大的元素;(c) 将式(11)中的增广矩阵分解成系数矩阵 $\boldsymbol{U}$ 和常数矩阵 $\boldsymbol{Y}$,形成上三角线性方程组 $\boldsymbol{UX} = \boldsymbol{Y}$。

(a) 构造增广矩阵为:

```
>> A=[1 2 1 4;2 0 4 3;4 2 2 1;-3 1 3 2];
>> B=[13 28 20 6]';
>> Aug=[A B]
Aug=
    1 2 1 4 13
    2 0 4 3 28
    4 2 2 1 20
   -3 1 3 2 6
```

(b) 在下面的 MATLAB 显示中,a 是矩阵 A 的第一列 1 中绝对值最大的元素,j 是行数:

```
>> [a,j] = max(abs(A(1:4,1)))
a =
     4
j =
     3
```

(c) 设 Augup = $[\boldsymbol{U} \mid \boldsymbol{Y}]$是(11)中的上三角矩阵。则有:

```
>> Augup = [1 2 1 4 13;0 -4 2 -5 2;0 0 -5 -7.5 -35;0 0 0 -9 18];
>> U = Augup(1:4,1:4)
U =
    1.0000  2.0000  1.0000   4.0000
     0     -4.0000  2.0000  -5.0000
     0       0     -5.0000  -7.5000
     0       0       0      -9.0000
>> Y = Augup(1:4,5)
Y =
      13
       2
     -35
     -18
```

程序 3.2 (上三角变换和回代过程)　为构造 $\boldsymbol{AX} = \boldsymbol{B}$ 的解,首先将增广矩阵$[\boldsymbol{A} \mid \boldsymbol{B}]$变换成上三角矩阵,再执行回代过程

```
function X = uptrbk(A,B)
% Input  - A is an N x N nonsingular matrix
%        - B is an N x 1 matrix
% Output - X is an N x 1 matrix containing the solution to AX = B.
% Initialize X and the temporary storage matrix C
    [N N] = size(A);
    X = zeros(N,1);
    C = zeros(1,N + 1);
% Form the augmented amtrix:Aug = [A|B]
    Aug = [A B];
for p = 1:N - 1
     % Partial pivoting for column p
    [Y,j] = max(abs(Aug(p:N,p)));
     % Interchange row p and j
    C = Aug(p,:);
    Aug(p,:) = Aug(j + p - 1,:);
    Aug(j + p - 1,:) = C;
if  Aug(p,p) == 0
    'A was singular. No unique solution'
    break
end
% Elimination process for column p
    for k = p + 1:N
      m = Aug(k,p)/Aug(p,p);
```

```
        Aug(k,p:N+1)=Aug(k,p:N+1)-m*Aug(p,p:N+1);
      end
    end
    % Back Substitution on [U|Y] using PRogram 3.1
    X=backsub(Aug(1:N,1:N),Aug(1:N,N+1));
```

3.4.5 高斯消去法和选主元的练习

在练习 1 到练习 4 中,证明线性方程组 $\boldsymbol{AX}=\boldsymbol{B}$ 等价于上三角线性方程组 $\boldsymbol{UX}=\boldsymbol{Y}$,并求解方程组。

1. $\begin{aligned} 2x_1+4x_2-6x_3&=-4 \\ x_1+5x_2+3x_3&=10 \\ x_1+3x_2+2x_3&=5 \end{aligned}$ $\qquad$ $\begin{aligned} 2x_1+4x_2-6x_3&=-4 \\ 3x_2+6x_3&=12 \\ 3x_3&=3 \end{aligned}$

2. $\begin{aligned} x_1+x_2+6x_3&=7 \\ -x_1+2x_2+9x_3&=2 \\ x_1-2x_2+3x_3&=10 \end{aligned}$ $\qquad$ $\begin{aligned} x_1+x_2+6x_3&=7 \\ 3x_2+15x_3&=9 \\ 12x_3&=12 \end{aligned}$

3. $\begin{aligned} 2x_1-2x_2+5x_3&=6 \\ 2x_1+3x_2+x_3&=13 \\ -x_1+4x_2-4x_3&=3 \end{aligned}$ $\qquad$ $\begin{aligned} 2x_1-2x_2+5x_3&=6 \\ 5x_2-4x_3&=7 \\ 0.9x_3&=1.8 \end{aligned}$

4. $\begin{aligned} -5x_1+2x_2-x_3&=-1 \\ x_1+0x_2+3x_3&=5 \\ 3x_1+x_2+6x_3&=17 \end{aligned}$ $\qquad$ $\begin{aligned} -5x_1+2x_2-x_3&=-1 \\ 0.4x_2+2.8x_3&=4.8 \\ -10x_3&=-10 \end{aligned}$

5. 求解抛物线 $y=A+Bx+Cx^2$ 的参数,抛物线经过点(1,4),(2,7)和(3,14)。
6. 求解抛物线 $y=A+Bx+Cx^2$ 的参数,抛物线经过点(1,6),(2,5),(3,2)。
7. 求解三次曲线 $y=A+Bx+Cx^2+Dx^3$ 的参数,三次曲线经过点(0,0),(1,1),(2,2),(3,2)。

在练习 8 到练习 10 中,证明线性方程组 $\boldsymbol{AX}=\boldsymbol{B}$ 与上三角线性方程组 $\boldsymbol{UX}=\boldsymbol{Y}$ 等价,并求解方程组:

8. $\begin{aligned} 4x_1+8x_2+4x_3+0x_4&=8 \\ x_1+5x_2+4x_3-3x_4&=-4 \\ x_1+4x_2+7x_3+2x_4&=10 \\ x_1+3x_2+0x_3-2x_4&=-4 \end{aligned}$ $\qquad$ $\begin{aligned} 4x_1+8x_2+4x_3+0x_4&=8 \\ 3x_2+3x_3-3x_4&=-6 \\ 4x_3+4x_4&=12 \\ x_4&=2 \end{aligned}$

9. $\begin{aligned} 2x_1+4x_2-4x_3+0x_4&=12 \\ x_1+5x_2-5x_3-3x_4&=18 \\ 2x_1+3x_2+x_3+3x_4&=8 \\ x_1+4x_2-2x_3+2x_4&=8 \end{aligned}$ $\qquad$ $\begin{aligned} 2x_1+4x_2-4x_3+0x_4&=12 \\ 3x_2-3x_3-3x_4&=12 \\ 4x_3+2x_4&=0 \\ 3x_4&=-6 \end{aligned}$

10. $\begin{aligned} x_1+2x_2+0x_3-x_4&=9 \\ 2x_1+3x_2-x_3+0x_4&=9 \\ 0x_1+4x_2+2x_3-5x_4&=26 \end{aligned}$ $\qquad$ $\begin{aligned} x_1+2x_2+0x_3-x_4&=9 \\ -x_2-x_3+2x_4&=-9 \\ -2x_3+3x_4&=-10 \end{aligned}$

$$5x_1+5x_2+2x_3-4x_4=32 \qquad 1.5x_4=-3$$

11. 求解下列线性方程组：

$$\begin{aligned} x_1+2x_2 \qquad\qquad &= 7\\ 2x_1+3x_2-x_3 \qquad &= 9\\ 4x_2+2x_3+3x_4 &= 10\\ 2x_3-4x_4 &= 12 \end{aligned}$$

12. 求解下列线性方程组：

$$\begin{aligned} x_1+x_2 \qquad\qquad &= 5\\ 2x_1-x_2+5x_3 \qquad &= -9\\ 3x_2-4x_3+2x_4 &= 19\\ 2x_3-6x_4 &= 2 \end{aligned}$$

13. Rockmore 公司考虑购买一台新的计算机，或者是 DoGood 174，或者是 MightDo 11。公司通过求解下列线性方程组，以测试计算机的能力：

$$\begin{aligned} 34x+55y-21&=0\\ 55x+89y-34&=0 \end{aligned}$$

DoGood 174 计算机的结果为 $x=-0.11$ 和 $y=0.45$，将它们代入方程组进行精确性检查得到：

$$\begin{aligned} 34(-0.11)+55(0.45)-21&=0.01\\ 55(-0.11)+89(0.45)-34&=0.00 \end{aligned}$$

MightDo 11 计算机的结果是 $x=-0.99$ 和 $y=1.01$，将它们代入方程组进行精确性检查得到：

$$\begin{aligned} 34(-0.99)+55(1.01)-21&=0.89\\ 55(-0.99)+89(1.01)-34&=1.44 \end{aligned}$$

哪一台计算机的答案更好？为什么？

14. 利用(i)偏序选主元策略的高斯消去法和(ii)按比例偏序选主元策略的高斯消去法，求解下列线性方程组：

(a)
$$\begin{aligned} 2x_1-3x_2+100x_3&=1\\ x_1+10x_2-0.001x_3&=0\\ 3x_1-100x_2+0.01x_3&=0 \end{aligned}$$

(b)
$$\begin{aligned} x_1+20x_2-x_3+0.001x_4&=0\\ 2x_1-5x_2+30x_3-0.1x_4&=1\\ 5x_1+x_2-100x_3-10x_4&=0\\ 2x_1-100x_2-x_3+x_4&=0 \end{aligned}$$

15. Hilbert 矩阵是一个典型的病态矩阵，如果对它的系数进行微小扰动，可对线性方程组的解产生极大的改变。

(a) 用 4×4 阶 Hilbert 矩阵求解 $\boldsymbol{AX}=\boldsymbol{B}$ 的精确解(用分数表示所有的元素并进行精确计算)：

$$A=\begin{bmatrix}1 & \frac{1}{2} & \frac{1}{3} & \frac{1}{4}\\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5}\\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \frac{1}{6}\\ \frac{1}{4} & \frac{1}{5} & \frac{1}{6} & \frac{1}{7}\end{bmatrix},\qquad B=\begin{bmatrix}1\\0\\0\\0\end{bmatrix}$$

(b) 使用精度为 4 位有效数字的算术计算求解 $AX=B$:

$$A=\begin{bmatrix}1.0000 & 0.5000 & 0.3333 & 0.2500\\ 0.5000 & 0.3333 & 0.2500 & 0.2000\\ 0.3333 & 0.2500 & 0.2000 & 0.1667\\ 0.2500 & 0.2000 & 0.1667 & 0.1429\end{bmatrix},\qquad B=\begin{bmatrix}1\\0\\0\\0\end{bmatrix}$$

注意:(b) 中的系数矩阵是(a) 中系数矩阵的近似值。

3.4.6 算法和程序

1. 许多科学应用包含的矩阵带有很多零。在实际情况中有如下形式的重要三角形线性方程组(参见练习 11 和练习 12):

$$\begin{aligned}
d_1x_1 + c_1x_2 \qquad\qquad\qquad\qquad &= b_1\\
a_1x_1 + d_2x_2 + c_2x_3 \qquad\qquad &= b_2\\
a_2x_2 + d_3x_3 + c_3x_4 \qquad &= b_3\\
\vdots \qquad\qquad &\quad \vdots\\
a_{N-2}x_{N-2} + d_{N-1}x_{N-1} + c_{N-1}x_N &= b_{N-1}\\
a_{N-1}x_{N-1} + d_Nx_N &= b_N
\end{aligned}$$

构造一个程序求解三角形线性方程组。可假定不需要行变换,而且可用第 k 行消去第 $k+1$ 行的 x_k。

2. 使用程序 3.2,求 6 次多项式 $y=a_1+a_2x+a_3x^2+a_4x^3+a_5x^4+a_6x^5+a_7x^6$ 的系数,它经过点(0,1),(1,3),(2,2),(3,1),(4,3),(5,2),(6,1)。使用 plot 命令画出多项式和上面所给定的经过点,并解释图中的偏差。
3. 使用程序 3.2 求解线性方程组 $AX=B$,其中 $A=[a_{ij}]_{N\times N}$,$a_{ij}=i^{j-1}$而且 $B=[b_{ij}]_{N\times 1}$,其中 $b_{11}=N$ 而且 $b_{i1}=(i^N-1)/(i-1)$或 $i\geqslant 2$。设 $N=3,7,11$,精确解为 $X=[1\quad 1\quad \ldots\quad 1\quad 1]'$。解释计算结果与精确解的偏差。
4. 构造一个程序,将程序 3.2 中的偏序选主元策略改成按比例偏序选主元策略。
5. 使用问题 4 中构造的按比例偏序选主元策略程序,求解问题 3 中 $N=11$ 的线性方程组。解释计算结果为何比问题 3 的计算结果好。
6. 修改程序 3.2,使得它能有效地求解具有相同系数矩阵 A 但常数矩阵B 不同的M 线性方程组集合。M 线性方程组集合如下所示:

$$\boldsymbol{AX}_1 = \boldsymbol{B}_1, \quad \boldsymbol{AX}_2 = \boldsymbol{B}_2, \quad \cdots, \quad \boldsymbol{AX}_M = \boldsymbol{B}_M$$

7. 下面的问题虽然是针对 3×3 阶矩阵，但其概念可用于 $N\times N$ 阶矩阵。如果矩阵 $\boldsymbol{A}$ 非奇异，则 $\boldsymbol{A}^{-1}$存在，而且 $\boldsymbol{AA}^{-1} = \boldsymbol{I}$。设 $\boldsymbol{C}_1, \boldsymbol{C}_2, \boldsymbol{C}_3$ 是 $\boldsymbol{A}^{-1}$的列，而 $\boldsymbol{E}_1, \boldsymbol{E}_2, \boldsymbol{E}_3$ 是 $\boldsymbol{I}$ 的列。方程 $\boldsymbol{AA}^{-1} = \boldsymbol{I}$ 可表示为：

$$\boldsymbol{A}[\boldsymbol{C}_1 \quad \boldsymbol{C}_2 \quad \boldsymbol{C}_3] = [\boldsymbol{E}_1 \quad \boldsymbol{E}_2 \quad \boldsymbol{E}_3]$$

则上式等价于三个线性方程组：

$$\boldsymbol{AC}_1 = \boldsymbol{E}_1, \quad \boldsymbol{AC}_2 = \boldsymbol{E}_2 \text{和} \ \boldsymbol{AC}_3 = \boldsymbol{E}_3$$

这样求 $\boldsymbol{A}^{-1}$等价于求解三个线性方程组。

使用程序 3.2 或问题 6 中的程序求解下面每个矩阵的逆。通过计算 $\boldsymbol{AA}^{-1}$和使用命令 inv(A)检查答案，并解释可能的差异：

$$\text{(a)}\begin{bmatrix} 2 & 0 & 1 \\ 3 & 2 & 5 \\ 1 & -1 & 0 \end{bmatrix} \qquad \text{(b)}\begin{bmatrix} 16 & -120 & 240 & -140 \\ -120 & 1200 & -2700 & 1680 \\ 240 & -2700 & 6480 & -4200 \\ -140 & 1680 & -4200 & 2800 \end{bmatrix}$$

3.5 三角分解法

在 3.3 节中，可以看到求解上三角线性方程组很容易。现在介绍将给定矩阵 $\boldsymbol{A}$ 分解成下三角矩阵 $\boldsymbol{L}$ 和上三角矩阵 $\boldsymbol{U}$ 的乘积的概念。这里下三角矩阵 $\boldsymbol{L}$ 的主对角线为 1，上三角矩阵 $\boldsymbol{U}$ 的对角线元素非零。为了便于表示，这里主要使用 4×4 阶矩阵表达各种概念，但这些概念也可用于任意 $N\times N$ 阶矩阵。

定义 3.4　如果非奇异矩阵 $\boldsymbol{A}$ 可表示为下三角矩阵 $\boldsymbol{L}$ 和上三角矩阵 $\boldsymbol{U}$ 的乘积：

$$\boldsymbol{A} = \boldsymbol{LU} \tag{1}$$

则 A 存在一个三角分解。

用矩阵形式可表示为：

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ m_{21} & 1 & 0 & 0 \\ m_{31} & m_{32} & 1 & 0 \\ m_{41} & m_{42} & m_{43} & 1 \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} & u_{13} & u_{14} \\ 0 & u_{22} & u_{23} & u_{24} \\ 0 & 0 & u_{33} & u_{34} \\ 0 & 0 & 0 & u_{44} \end{bmatrix}$$

矩阵 $\boldsymbol{A}$ 非奇异的条件为对所有 k 有 $u_{kk}\neq0$。$\boldsymbol{L}$ 中元素表示为 m_{ij}。后面将解释选择 m_{ij} 而不是 l_{ij} 的原因。

3.5.1 线性方程组的解

设线性方程组 $\boldsymbol{AX} = \boldsymbol{B}$ 的系数矩阵 $\boldsymbol{A}$ 存在三角分解式(1)，则线性方程组可表示为：

$$\boldsymbol{LUX} = \boldsymbol{B} \tag{2}$$

而方程组的解可通过定义 $\boldsymbol{Y} = \boldsymbol{UX}$ 并求解下面的两个方程组得到。

首先对方程组 $\boldsymbol{LY} = \boldsymbol{B}$ 求解 $\boldsymbol{Y}$；然后对方程组 $\boldsymbol{UX} = \boldsymbol{Y}$ 求解 $\boldsymbol{X}$　(3)

必须首先求解下三角线性方程组：

$$\begin{aligned} y_1 &&&&= b_1 \\ m_{21}y_1 + && y_2 &&= b_2 \\ m_{31}y_1 + m_{32}y_2 + && y_3 &&= b_3 \\ m_{41}y_1 + m_{42}y_2 + m_{43}y_3 + y_4 &&&&= b_4 \end{aligned} \tag{4}$$

得到 y_1, y_2, y_3, y_4。然后使用它们求解上三角线性方程组：

$$\begin{aligned} u_{11}x_1 + u_{12}x_2 + u_{13}x_3 + u_{14}x_4 &= y_1 \\ u_{22}x_2 + u_{23}x_3 + u_{24}x_4 &= y_2 \\ u_{33}x_3 + u_{34}x_4 &= y_3 \\ u_{44}x_4 &= y_4 \end{aligned} \tag{5}$$

例 3.20 求解：

$$\begin{aligned} x_1 + 2x_2 + 4x_3 + x_4 &= 21 \\ 2x_1 + 8x_2 + 6x_3 + 4x_4 &= 52 \\ 3x_1 + 10x_2 + 8x_3 + 8x_4 &= 79 \\ 4x_1 + 12x_2 + 10x_3 + 6x_4 &= 82 \end{aligned}$$

解：

利用三角分解法和如下等式，可得：

$$\boldsymbol{A} = \begin{bmatrix} 1 & 2 & 4 & 1 \\ 2 & 8 & 6 & 4 \\ 3 & 10 & 8 & 8 \\ 4 & 12 & 10 & 6 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 3 & 1 & 1 & 0 \\ 4 & 1 & 2 & 1 \end{bmatrix} \begin{bmatrix} 1 & 2 & 4 & 1 \\ 0 & 4 & -2 & 2 \\ 0 & 0 & -2 & 3 \\ 0 & 0 & 0 & -6 \end{bmatrix} = \boldsymbol{LU}$$

使用前向替换法求解 $\boldsymbol{LY} = \boldsymbol{B}$：

$$\begin{aligned} y_1 &&&= 21 \\ 2y_1 + y_2 &&&= 52 \\ 3y_1 + y_2 + y_3 &&&= 79 \\ 4y_1 + y_2 + 2y_3 + y_4 &&&= 82 \end{aligned} \tag{6}$$

得到值 $y_1 = 21, y_2 = 52 - 2(21) = 10, y_3 = 79 - 3(21) - 10 = 6, y_4 = 82 - 4(21) - 10 - 2(6) = -24$，或表示为 $\boldsymbol{Y} = [21 \quad 10 \quad 6 \quad -24]'$。接下来将方程组 $\boldsymbol{UX} = \boldsymbol{Y}$ 表示为：

$$\begin{aligned} x_1 + 2x_2 + 4x_3 + x_4 &= 21 \\ 4x_2 - 2x_3 + 2x_4 &= 10 \\ -2x_3 + 3x_4 &= 6 \\ -6x_4 &= -24 \end{aligned} \tag{7}$$

现在利用回代法可得到值 $x_4 = -24/(-6) = 4, x_3 = (6 - 3(4))/(-2) = 3, x_2 = (10 - 2(4) + 2(3))/4 = 2, x_1 = 21 - 4 - 4(3) - 2(2) = 1$，或表示为 $\boldsymbol{X} = [1 \quad 2 \quad 3 \quad 4]'$。

3.5.2 三角分解法

现在讨论如何得到矩阵的三角分解。当使用高斯消去法时，如果行交换变换不是必须的，

则倍数 m_{ij} 是 $\boldsymbol{L}$ 中的子对角线元素。

例 3.21　使用高斯消去法构造下列矩阵的三角分解：

$$\boldsymbol{A}=\begin{bmatrix}4 & 3 & -1\\ -2 & -4 & 5\\ 1 & 2 & 6\end{bmatrix}$$

解：

通过将单位矩阵 I 放在 A 的左边来构造矩阵 $\boldsymbol{L}$。对每个用来构造上三角矩阵行变换，将倍数 m_{ij} 放在左边的对应位置。初始矩阵为：

$$\boldsymbol{A}=\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}4 & 3 & -1\\ -2 & -4 & 5\\ 1 & 2 & 6\end{bmatrix}$$

用行 1 消去矩阵 $\boldsymbol{A}$ 的列 1 中 a_{11} 下面的元素。行 2 和行 3 分别减去行 1 乘倍数 $m_{21}=-0.5$ 和 $m_{31}=0.25$。将倍数放到矩阵的左边相应位置，结果为：

$$\boldsymbol{A}=\begin{bmatrix}1 & 0 & 0\\ -0.5 & 1 & 0\\ 0.25 & 0 & 1\end{bmatrix}\begin{bmatrix}4 & 3 & -1\\ 0 & -2.5 & 4.5\\ 0 & 1.25 & 6.25\end{bmatrix}$$

用行 2 消去列 2 中对角线下的元素。行 3 减去行 2 乘倍数 $m_{32}=-0.5$，再将倍数放入矩阵左边，则可得到矩阵 $\boldsymbol{A}$ 的三角分解：

$$\boldsymbol{A}=\begin{bmatrix}1 & 0 & 0\\ -0.5 & 1 & 0\\ 0.25 & -0.5 & 1\end{bmatrix}\begin{bmatrix}4 & 3 & -1\\ 0 & -2.5 & 4.5\\ 0 & 0 & 8.5\end{bmatrix}\tag{8}$$

定理 3.10（A = LU 的直接分解，无行交换变换）　设无行交换变换的高斯消去法可求解一般线性方程组 $\boldsymbol{AX}=\boldsymbol{B}$，则矩阵 $\boldsymbol{A}$ 可分解为一个下三角矩阵 $\boldsymbol{L}$ 和一个上三角矩阵 $\boldsymbol{U}$ 的乘积：

$$\boldsymbol{A}=\boldsymbol{LU}$$

而且 $\boldsymbol{L}$ 的对角线元素为 1，$\boldsymbol{U}$ 的对角线元素非零。得到 $\boldsymbol{L}$ 和 $\boldsymbol{U}$ 后，可通过如下步骤得到 $\boldsymbol{X}$：

1. 利用前向替换法对方程组 $\boldsymbol{LY}=\boldsymbol{B}$ 求解 $\boldsymbol{Y}$
2. 利用回代法对方程组 $\boldsymbol{UX}=\boldsymbol{Y}$ 求解 $\boldsymbol{X}$

证明：当执行高斯消去过程，并将 $\boldsymbol{B}$ 存入增广矩阵中（增广矩阵有 $N+1$ 列）时，上三角分解处理后的结果是等价的上三角线性方程组 $\boldsymbol{UX}=\boldsymbol{Y}$。矩阵 $\boldsymbol{L},\boldsymbol{U},\boldsymbol{B},\boldsymbol{Y}$ 有如下形式：

$$\boldsymbol{L}=\begin{bmatrix}1 & 0 & 0 & & 0\\ m_{21} & 1 & 0 & \cdots & 0\\ m_{31} & m_{32} & 1 & \cdots & 0\\ \vdots & \vdots & \vdots & & \vdots\\ m_{N1} & m_{N2} & m_{N3} & \cdots & 1\end{bmatrix},\quad \boldsymbol{B}=\begin{bmatrix}a_{1\,N+1}^{(1)}\\ a_{2\,N+1}^{(2)}\\ a_{3\,N+1}^{(3)}\\ \vdots\\ a_{N\,N+1}^{(N)}\end{bmatrix}$$

$$U=\begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} & \cdots & a_{1N}^{(1)} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & \cdots & a_{2N}^{(2)} \\ 0 & 0 & a_{33}^{(3)} & \cdots & a_{3N}^{(3)} \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \cdots & a_{NN}^{(N)} \end{bmatrix}, \quad Y=\begin{bmatrix} a_{1\,N+1}^{(1)} \\ a_{2\,N+1}^{(2)} \\ a_{3\,N+1}^{(3)} \\ \vdots \\ a_{N\,N+1}^{(N)} \end{bmatrix}$$

注：如果只寻找 $\boldsymbol{L}$ 和 $\boldsymbol{U}$,可不需要增广矩阵的第 $N+1$ 列。

第 1 步:将系数存入增广矩阵。$a_{rc}^{(1)}$ 的上标表示位于矩阵(r,c)处的值是第一次存放的:

$$\left[\begin{array}{ccccc|c} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} & \cdots & a_{1N}^{(1)} & a_{1\,N+1}^{(1)} \\ a_{21}^{(1)} & a_{22}^{(1)} & a_{23}^{(1)} & \cdots & a_{2N}^{(1)} & a_{2\,N+1}^{(1)} \\ a_{31}^{(1)} & a_{32}^{(1)} & a_{33}^{(1)} & \cdots & a_{3N}^{(1)} & a_{3\,N+1}^{(1)} \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ a_{N1}^{(1)} & a_{N2}^{(1)} & a_{N3}^{(1)} & \cdots & a_{NN}^{(1)} & a_{N\,N+1}^{(1)} \end{array}\right]$$

第 2 步:消去行 2 到行 N 中的 x_1,并将用于消去行 r 中的 x_1 的倍数 m_{r1} 存入矩阵$(r,1)$处:

for $r=2:N$

  $m_{r1}=a_{r1}^{(1)}/a_{11}^{(1)}$;

  $a_{r1}=m_{r1}$;

  for $c=2:N+1$

    $a_{rc}^{(2)}=a_{rc}^{(1)}-m_{r1}*a_{1c}^{(1)}$;

  end

end

新的元素写成 $a_{rc}^{(2)}$,表示在矩阵中的位置(r,c)处第二次存放的值。执行步骤 2 后的结果为:

$$\left[\begin{array}{ccccc|c} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} & \cdots & a_{1N}^{(1)} & a_{1\,N+1}^{(1)} \\ m_{21} & a_{22}^{(2)} & a_{23}^{(2)} & \cdots & a_{2N}^{(2)} & a_{2\,N+1}^{(2)} \\ m_{31} & a_{32}^{(2)} & a_{33}^{(2)} & \cdots & a_{3N}^{(2)} & a_{3\,N+1}^{(2)} \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ m_{N1} & a_{N2}^{(2)} & a_{N3}^{(2)} & \cdots & a_{NN}^{(2)} & a_{N\,N+1}^{(2)} \end{array}\right]$$

第 3 步:消去行 3 到行 N 中的 x_2,并将用于消去行 r 中 x_2 的倍数 m_{r2}存入矩阵$(r,2)$处:

for $r=3:N$

  $m_{r2}=a_{r2}^{(2)}/a_{22}^{(2)}$;

  $a_{r2}=m_{r2}$;

  for $c=3:N+1$

    $a_{rc}^{(3)}=a_{rc}^{(2)}-m_{r2}*a_{2c}^{(2)}$;

  end

end

新的元素写成 $a_{rc}^{(3)}$，表示在矩阵中的位置(r,c)处第三次存放的值。执行步骤 3 后的结果为：

$$\left[\begin{array}{ccccc|c} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} & \cdots & a_{1N}^{(1)} & a_{1\,N+1}^{(1)} \\ m_{21} & a_{22}^{(2)} & a_{23}^{(2)} & \cdots & a_{2N}^{(2)} & a_{2\,N+1}^{(2)} \\ m_{31} & m_{32} & a_{33}^{(3)} & \cdots & a_{3N}^{(3)} & a_{3\,N+1}^{(3)} \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ m_{N1} & m_{N2} & a_{N3}^{(3)} & \cdots & a_{NN}^{(3)} & a_{N\,N+1}^{(3)} \end{array}\right]$$

第 $p+1$ 步：这是一般步骤。消去行 $p+1$ 到行 N 中的 x_p，并将用于消去行 r 中的 x_p 的倍数存入矩阵(r,p)处：

```
for r = p + 1:N
    m_rp = a_rp^(p) / a_pp^(p);
    a_rp = m_rp;
    for c = p + 1:N + 1
        a_rc^(p+1) = a_rc^(p) - m_rp * a_pc^(p);
    end
end
```

将行 N 中的 x_{N-1} 消去后得到的最终结果是：

$$\left[\begin{array}{ccccc|c} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} & \cdots & a_{1N}^{(1)} & a_{1\,N+1}^{(1)} \\ m_{21} & a_{22}^{(2)} & a_{23}^{(2)} & \cdots & a_{2N}^{(2)} & a_{2\,N+1}^{(2)} \\ m_{31} & m_{32} & a_{33}^{(3)} & \cdots & a_{3N}^{(3)} & a_{3\,N+1}^{(3)} \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ m_{N1} & m_{N2} & m_{N3} & \cdots & a_{NN}^{(N)} & a_{N\,N+1}^{(N)} \end{array}\right]$$

上三角处理过程执行完毕。要注意只使用了一个数组保存 $\boldsymbol{L}$ 和 $\boldsymbol{U}$ 中的元素。$\boldsymbol{L}$ 中的对角线元素 1 没有保存，而且 $\boldsymbol{L}$ 中位于对角线上的元素 0 和 $\boldsymbol{U}$ 中位于对角线下的元素 0 也没有保存。只有用来重构 $\boldsymbol{L}$ 和 $\boldsymbol{U}$ 的系数元素才被保存。

现在验证 $\boldsymbol{LU} = \boldsymbol{A}$。设 $\boldsymbol{D} = \boldsymbol{LU}$，而且当 $r \leqslant c$ 时，d_{rc}表示为：

$$d_{rc} = m_{r1}a_{1c}^{(1)} + m_{r2}a_{2c}^{(2)} + \cdots + m_{rr-1}a_{r-1c}^{(r-1)} + a_{rc}^{(r)} \tag{9}$$

使用步骤 1 到步骤 $p+1=r$ 中的替换方程，可得到如下替换：

$$\begin{aligned} m_{r1}a_{1c}^{(1)} &= a_{rc}^{(1)} - a_{rc}^{(2)}, \\ m_{r2}a_{2c}^{(2)} &= a_{rc}^{(2)} - a_{rc}^{(3)}, \\ &\vdots \\ m_{rr-1}a_{r-1c}^{(r-1)} &= a_{rc}^{(r-1)} - a_{rc}^{(r)} \end{aligned} \tag{10}$$

将式(10)中的表达式代入式(9)中，可得到：

$$d_{rc} = a_{rc}^{(1)} - a_{rc}^{(2)} + a_{rc}^{(2)} - a_{rc}^{(3)} + \cdots + a_{rc}^{(r-1)} - a_{rc}^{(r)} + a_{rc}^{(r)} = a_{rc}^{(1)}$$

对于其他情况，即 $r > c$ 时，可进行类似的证明。

3.5.3 计算复杂性

高斯消去法和三角分解法的三角化过程是一样的。如果观察定理 3.10 中增广矩阵的前 N 列,则可得出方法的计算次数。

第 $p+1$ 步的外层循环需要 $N-p=N-(p+1)+1$ 次除法来得到倍数 m_{rp}。而在循环内,对前 N 列,需要 $(N-p)(N-p)$ 次乘法和相同次数的减法来得到新的行元素 $a_{rc}^{(p+1)}$。这个过程对 $p=1,2,\ldots,N-1$ 都要执行。这样 $\boldsymbol{A}=\boldsymbol{LU}$ 的三角分解部分需要:

$$\sum_{p=1}^{N-1}(N-p)(N-p+1)=\frac{N^3-N}{3}\quad \text{次乘法和除法} \tag{11}$$

以及:

$$\sum_{p=1}^{N-1}(N-p)(N-p)=\frac{2N^3-3N^2+N}{6}\quad \text{次减法} \tag{12}$$

通过利用下列求和公式可得到式(11):

$$\sum_{k=1}^{M}k=\frac{M(M+1)}{2}\quad \text{和}\quad \sum_{k=1}^{M}k^2=\frac{M(M+1)(2M+1)}{6}$$

使用变量 $k=N-p$ 重写式(11)可得:

$$\begin{aligned}\sum_{p=1}^{N-1}(N-p)(N-p+1)&=\sum_{p=1}^{N-1}(N-p)+\sum_{p=1}^{N-1}(N-p)^2\\&=\sum_{k=1}^{N-1}k+\sum_{k=1}^{N-1}k^2\\&=\frac{(N-1)N}{2}+\frac{(N-1)(N)(2N-1)}{6}\\&=\frac{N^3-N}{3}\end{aligned}$$

当得到 $\boldsymbol{A}=\boldsymbol{LU}$ 的三角分解后,接下来求解下三角线性方程组 $\boldsymbol{LY}=\boldsymbol{B}$ 将需要 $0+1+\cdots+N-1=(N^2-N)/2$ 次乘法和减法,但不需要除法,因为 $\boldsymbol{L}$ 的对角元素为 1。然后求解上三角线性方程组 $\boldsymbol{UX}=\boldsymbol{Y}$ 需要 $1+2+\cdots+N=(N^2+N)/2$ 次乘法和除法,还需要 $(N^2-N)/2$ 次减法。这样求解 $\boldsymbol{LUX}=\boldsymbol{B}$ 需要:

$$N^2\ \text{次乘法与除法,以及}\ N^2-N\ \text{次减法}$$

可以看到整个求解过程中三角分解部分占了主要的计算量。如果对线性方程组求解多次,而每次的线性方程组的系数矩阵 $\boldsymbol{A}$ 相同,列矩阵 $\boldsymbol{B}$ 不同,则如果保存了三角分解因子,就没有必要每次进行三角分解了。这也是通常选用三角分解法而不是消去法的原因。然而如果只求解一个线性方程组,则除了三角分解法要保存倍数外,两个方法是一样的。

3.5.4 置换矩阵

定理 3.10 中 $\boldsymbol{A}=\boldsymbol{LU}$ 的三角分解假设不存在行交换。这可能使得一个非奇异矩阵 $\boldsymbol{A}$ 不能直接分解为 $\boldsymbol{A}=\boldsymbol{LU}$。

例 3.22 证明下列矩阵不能直接分解为 $\boldsymbol{A}=\boldsymbol{LU}$:

$$
\boldsymbol{A}=\begin{bmatrix} 1 & 2 & 6 \\ 4 & 8 & -1 \\ -2 & 3 & 5 \end{bmatrix}
$$

证明:

设 $\boldsymbol{A}$ 存在一个直接 $\boldsymbol{LU}$ 分解,则:

$$
\begin{bmatrix} 1 & 2 & 6 \\ 4 & 8 & -1 \\ -2 & 3 & 5 \end{bmatrix}=\begin{bmatrix} 1 & 0 & 0 \\ m_{21} & 1 & 0 \\ m_{31} & m_{32} & 1 \end{bmatrix}\begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{bmatrix} \tag{13}
$$

式(13)中右边的矩阵 $\boldsymbol{L}$ 和 $\boldsymbol{U}$ 相乘并与对应的矩阵 $\boldsymbol{A}$ 中的元素进行比较。在列 1 中,$1=1u_{11}$,然后,$4=m_{21}u_{11}=m_{21}$,最后,$-2=m_{31}u_{11}=m_{31}$。在列 2 中,$2=1u_{12}$,然后,$8=m_{21}u_{12}=(4)(2)+u_{22}$,这表示 $u_{22}=0$,最后,$3=m_{31}u_{12}+m_{32}u_{22}=(-2)(2)+m_{32}(0)=-4$,这里产生了矛盾。所以 $\boldsymbol{A}$ 没有一个 $\boldsymbol{LU}$ 分解。

前 N 个正整数 $1,2,\cdots,N$ 的一个置换是这些数的一个确定顺序的排列 $k_1,k_2,\cdots,k_N$。例如,1,4,2,3,5 是 5 个整数 1,2,3,4,5 的一个置换。

在下面的定义中将使用标准基向量 $\boldsymbol{E}_i=[0\ 0\ldots\ 0\ 1_i\ 0\ldots 0]$,$i=1,2,\cdots,N$。

定义 3.5 一个 $N\times N$ 置换矩阵 $\boldsymbol{P}$ 是一个在每一行和每一列只有一个元素为 1,而其他元素为 0 的矩阵。$\boldsymbol{P}$ 的行是单位矩阵行的置换,可表示为:

$$
\boldsymbol{P}=[\boldsymbol{E}'_{k1}\quad \boldsymbol{E}'_{k2}\cdots\ \boldsymbol{E}'_{kN}]' \tag{14}
$$

$\boldsymbol{P}=[p_{ij}]$的元素有如下形式:

$$
p_{ij}=\begin{cases}1 & j=k_i \\ 0 & \text{其他情况}\end{cases}
$$

例如,下列 4×4 矩阵是一个置换矩阵:

$$
\boldsymbol{P}=\begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}=[\boldsymbol{E}'_2\quad \boldsymbol{E}'_1\quad \boldsymbol{E}'_4\quad \boldsymbol{E}'_3]' \tag{15}
$$

定理 3.11 设 $\boldsymbol{P}=[\boldsymbol{E}'_{k1}\quad \boldsymbol{E}'_{k2}\quad\cdots\quad \boldsymbol{E}'_{kN}]'$是一个置换矩阵。$\boldsymbol{PA}$ 是一个新的矩阵,它的行是将 $\boldsymbol{A}$ 中的行按行 $k_1\boldsymbol{A}$,行 $k_2\boldsymbol{A}$,$\cdots$,行 $k_N\boldsymbol{A}$ 调整顺序后形成的。

例 3.23 设 $\boldsymbol{A}$ 为 4×4 矩阵,$\boldsymbol{P}$ 为式(15)中的置换矩阵,则 $\boldsymbol{PA}$ 矩阵中的行是将 $\boldsymbol{A}$ 中的行调整顺序后形成的,顺序为第 1,2,3,4 行对应于 $\boldsymbol{A}$ 中的第 2,1,4,3 行。

解:

通过计算矩阵乘积,可得:

$$
\begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix}=\begin{bmatrix} a_{21} & a_{22} & a_{23} & a_{24} \\ a_{11} & a_{12} & a_{13} & a_{14} \\ a_{41} & a_{42} & a_{43} & a_{44} \\ a_{31} & a_{32} & a_{33} & a_{34} \end{bmatrix}
$$

定理 3.12 如果 $\boldsymbol{A}$ 是非奇异矩阵,则存在一个置换矩阵 $\boldsymbol{P}$,使得 $\boldsymbol{PA}$ 存在三角分解:

$$\boldsymbol{PA} = \boldsymbol{LU} \tag{16}$$

定理的证明可参见高等线性代数教材。

例 3.24 如果将例 3.22 中矩阵的行 2 和行 3 进行互换,则得到的 $\boldsymbol{PA}$ 有一个三角分解。

互换行 2 和行 3 的置换矩阵为 $\boldsymbol{P} = [\boldsymbol{E}'_1 \quad \boldsymbol{E}'_3 \quad \boldsymbol{E}'_2]'$。计算 $\boldsymbol{PA}$ 的乘积可得:

$$\boldsymbol{PA} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 2 & 6 \\ 4 & 8 & -1 \\ -2 & 3 & 5 \end{bmatrix} = \begin{bmatrix} 1 & 2 & 6 \\ -2 & 3 & 5 \\ 4 & 8 & -1 \end{bmatrix}$$

现在利用不带行交换的高斯消去法可得:

$$\begin{matrix} \text{主元}\rightarrow \\ m_{21} = -2 \\ m_{31} = \ \ 4 \end{matrix} \begin{bmatrix} \underline{1} & 2 & 6 \\ -2 & 3 & 5 \\ 4 & 8 & -1 \end{bmatrix}$$

从行 2 和列 3 中消去 x_2 后,可得:

$$\begin{matrix} \\ \text{主元}\rightarrow \\ m_{32} = 0 \end{matrix} \begin{bmatrix} 1 & 2 & 6 \\ 0 & \underline{7} & 17 \\ 0 & 0 & -25 \end{bmatrix} = \boldsymbol{U}$$

3.5.5 扩展高斯消去过程

下面的定理是定理 3.10 的扩展,它包含对行交换的处理。这样三角分解法可用于任何 $\boldsymbol{A}$ 是非奇异矩阵的线性方程组 $\boldsymbol{AX} = \boldsymbol{B}$。

定理 3.14 (非直接分解: $\boldsymbol{PA} = \boldsymbol{LU}$) 设 $\boldsymbol{A}$ 是一 $N \times N$ 矩阵。假设高斯消去法可求解经过行变换的一般线性方程组 $\boldsymbol{AX} = \boldsymbol{B}$。则存在一个置换矩阵 $\boldsymbol{P}$,使得 $\boldsymbol{PA}$ 可分解为一个下三角矩阵 $\boldsymbol{L}$ 和一个上三角矩阵 $\boldsymbol{U}$:

$$\boldsymbol{PA} = \boldsymbol{LU}$$

而且可构造 $\boldsymbol{L}$ 的主对角线元素为 1,$\boldsymbol{U}$ 的主对角线元素非零。可用如下 4 步求出 $\boldsymbol{X}$:

1. 构造矩阵 $\boldsymbol{L}, \boldsymbol{U}, \boldsymbol{P}$
2. 计算列向量 $\boldsymbol{PB}$
3. 用前向替换法对方程组 $\boldsymbol{LY} = \boldsymbol{PB}$ 求解 $\boldsymbol{Y}$
4. 用回代法对方程组 $\boldsymbol{UX} = \boldsymbol{Y}$ 求解 $\boldsymbol{X}$

注:如果要求解多个方程组 $\boldsymbol{AX} = \boldsymbol{B}$,其中矩阵 $\boldsymbol{A}$ 固定,而列矩阵 $\boldsymbol{B}$ 可变。则步骤 1 只需要执行一次,步骤 2 到步骤 4 根据不同的 $\boldsymbol{B}$ 求解 $\boldsymbol{X}$。求解 $\boldsymbol{X}$ 的步骤 2 到步骤 4 的计算效率很高,需要 $O(N^2)$ 次操作,而高斯消去法需要 $O(N^3)$ 次操作。

3.5.6 MATLAB

MATLAB 命令[L,U,P] = lu(A)可得到下三角矩阵 $\boldsymbol{L}$ 和上三角矩阵 $\boldsymbol{U}$(通过对 $\boldsymbol{A}$ 进行三角分解),以及定理 3.14 中的置换矩阵 $\boldsymbol{P}$。

例 3.25 对练习 3.22 中的矩阵 A 使用 MATLAB 命令[L,U,P] = lu(A)。验证 $A = P^{-1}LU$(即证明 $PA = LU$):

```
>>A=[1 2 6 ;4 8 -1;-2 3 -5];
>>[L,U,P]=lu(A)
L=
   1.0000   0        0
   -0.5000  1.0000   0
   0.2500   0        1.0000
U=
   4.0000   8.0000  -1.0000
   0        7.0000   4.5000
   0        0        6.2500
P=
   0 1 0
   0 0 1
   1 0 0

>>inv(P)*L*U
   1  2  6
   4  8 -1
  -2 3  5
```

正如前面所指出的,研究人员经常使用的是三角分解法而不是消去法。在 MATLAB 中的 inv(A)和 det(A)也利用三角分解法。例如,根据线性代数的理论,可知道非奇异矩阵 A 的行列式等于 $(-1)^q \det U$,这里的 U 是矩阵 A 三角分解产生的上三角矩阵,而 q 是从单位矩阵 I 得到 P 所交换的行的次数。由于 U 是上三角矩阵,所以 U 的行列式是它主对角线元素的乘积(参见定理 3.6)。读者可以对例 3.25 进行验证,即 $\det(A) = 175 = (-1)^2(175) = (-1)^2\det(U)$。

下面的程序实现了定理 3.10 的证明中描述的处理过程。它是程序 3.2 的扩展,并使用部分选主元策略。由偏序选主元带来的行交换记录在矩阵 R 中。然后在前向替换步骤中使用矩阵 R 求解矩阵 Y。

程序 3.3 (PA = LU:带选主元的分解法) 构造线性方程组 $AX = B$ 的解,这里 A 是非奇异矩阵

```
function X = lufact(A,B)
% Input  - A is an N x N matrix
%        - B is an N x 1 matrix
% Output - X is an N x 1 matrix containing the solution to AX = B.
% Initialize X,Y,the temporary storage matrix C,and the row
% permutation information matrix R
  [N,N]=size(A);
  X=zeros(N,1);
  Y=zeros(N,1);
  C=zeros(1,N);
  R=1:1;

for p=1:N-1
```

```
% Find the pivot row for column p
   [max1,j] = max(abs(A(p:N,p)));
% Interchange row p and j
   C = A(p,:);
   A(p,:) = A(j + p - 1,:);
   A(j + p - 1,:) = C;
   d = R(p);
   R(p) = R(j + p - 1);
   R(j + p - 1) = d;
if A(p,p) == 0
   'A is singular. No unique solution'
   break
end
% Calculate multiplier and place in subdiagonal portion of A
   for k = p + 1:N
      mult = A(k,p)/A(p,p);
      A(k,p) = mult;
      A(k,p + 1:N) = A(k,p + 1:N) - mult * A(p,P + 1:N);
   end
end
% Solve for Y
Y(1) = B(R(1));
for k = 2:N
   Y(k) = B(R(k)) - A(k,1:k - 1) * Y(1:k - 1)
end
% Solve for X
X(N) = Y(N)/A(N,N);
for k = N - 1: - 1:1
   X(k) = (Y(k) - A(k,k + 1:N) * X(k + 1:N))/A(k,k);
end
```

3.5.7 三角分解法的练习

1. (a) $\boldsymbol{B}=[-4 \quad 10 \quad 5]'$ (b) $\boldsymbol{B}=[20 \quad 49 \quad 32]'$, $\boldsymbol{A}=\boldsymbol{LU}$ 表示为:

$$\begin{bmatrix} 2 & 4 & -6 \\ 1 & 5 & 3 \\ 1 & 3 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 1/2 & 1 & 0 \\ 1/2 & 1/3 & 1 \end{bmatrix} \begin{bmatrix} 2 & 4 & -6 \\ 0 & 3 & 6 \\ 0 & 0 & 3 \end{bmatrix}$$

求解 $\boldsymbol{LY}=\boldsymbol{B}$, $\boldsymbol{UX}=\boldsymbol{Y}$,并验证 $\boldsymbol{B}=\boldsymbol{AX}$。

2. (a) $\boldsymbol{B}=[7 \quad 2 \quad 10]'$ (b) $\boldsymbol{B}=[23 \quad 35 \quad 7]'$, $\boldsymbol{A}=\boldsymbol{LU}$ 表示为:

$$\begin{bmatrix} 1 & 1 & 6 \\ -1 & 2 & 9 \\ 1 & -2 & 3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 1 & -1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 6 \\ 0 & 3 & 15 \\ 0 & 0 & 12 \end{bmatrix}$$

求解 $\boldsymbol{LY}=\boldsymbol{B}$, $\boldsymbol{UX}=\boldsymbol{Y}$,并验证 $\boldsymbol{B}=\boldsymbol{AX}$。

3. 对下列矩阵求解它的三角分解 $\boldsymbol{L}$ 和 $\boldsymbol{U}$。

(a) $\begin{bmatrix} -5 & 2 & -1 \\ 1 & 0 & 3 \\ 3 & 1 & 6 \end{bmatrix}$ (b) $\begin{bmatrix} 1 & 0 & 3 \\ 3 & 1 & 6 \\ -5 & 2 & -1 \end{bmatrix}$

4．对下列矩阵求解它的三角分解 $\boldsymbol{L}$ 和 $\boldsymbol{U}$。

(a) $\begin{bmatrix} 4 & 2 & 1 \\ 2 & 5 & -2 \\ 1 & -2 & 7 \end{bmatrix}$　　(b) $\begin{bmatrix} 1 & -2 & 7 \\ 4 & 2 & 1 \\ 2 & 5 & -2 \end{bmatrix}$

5．(a) $\boldsymbol{B} = [8 \quad -4 \quad 10 \quad -4]'$　(b) $\boldsymbol{B} = [28 \quad 13 \quad 23 \quad 4]'$

$\boldsymbol{A} = \boldsymbol{LU}$ 表示为：

$$\begin{bmatrix} 4 & 8 & 4 & 0 \\ 1 & 5 & 4 & -3 \\ 1 & 4 & 7 & 2 \\ 1 & 3 & 0 & -2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{1}{4} & 1 & 0 & 0 \\ \frac{1}{4} & \frac{2}{3} & 1 & 0 \\ \frac{1}{4} & \frac{1}{3} & -\frac{1}{2} & 1 \end{bmatrix} \begin{bmatrix} 4 & 8 & 4 & 0 \\ 0 & 3 & 3 & -3 \\ 0 & 0 & 4 & 4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

求解 $\boldsymbol{LY} = \boldsymbol{B}$，$\boldsymbol{UX} = \boldsymbol{Y}$，并验证 $\boldsymbol{B} = \boldsymbol{AX}$。

6．对下列矩阵求解它的三角分解 $\boldsymbol{L}$ 和 $\boldsymbol{U}$。

$$\begin{bmatrix} 1 & 1 & 0 & 4 \\ 2 & -1 & 5 & 0 \\ 5 & 2 & 1 & 2 \\ -3 & 0 & 2 & 6 \end{bmatrix}$$

7．试推导出式(12)中的公式。

8．证明在下面的情况下三角分解是惟一的：如果矩阵 $\boldsymbol{A}$ 非奇异，而且 $\boldsymbol{L}_1\boldsymbol{U}_1 = \boldsymbol{A} = \boldsymbol{L}_2\boldsymbol{U}_2$，则 $\boldsymbol{L}_1 = \boldsymbol{L}_2$，且 $\boldsymbol{U}_1 = \boldsymbol{U}_2$。

9．证明定理 3.10 中 $r > c$ 的情况。

10．(a) 通过对下列置换矩阵证明 $\boldsymbol{PP}' = \boldsymbol{I} = \boldsymbol{P}'\boldsymbol{P}$ 来验证定理 3.12。

$$\boldsymbol{P} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

(b) 证明定理 3.12。提示：利用矩阵乘定义，以及 $\boldsymbol{P}$ 与 $\boldsymbol{P}'$ 的每行和每列只有一个元素为 1 的事实。

11．证明一个 $N \times N$ 上三角矩阵的逆也是一个上三角矩阵。

3.5.8　算法和程序

1．使用程序 3.3 求解线性方程组 $\boldsymbol{AX} = \boldsymbol{B}$，其中：

$$\boldsymbol{A} = \begin{bmatrix} 1 & 3 & 5 & 7 \\ 2 & -1 & 3 & 5 \\ 0 & 0 & 2 & 5 \\ -2 & -6 & -3 & 1 \end{bmatrix} \quad 和 \quad \boldsymbol{B} = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}$$

使用 MATLAB 中的[L,U,P] = lu(A)命令检查得到的答案。

2．使用程序 3.3 求解线性方程组 $\boldsymbol{AX} = \boldsymbol{B}$，其中 $\boldsymbol{A} = [a_{ij}]_{N\times N}$，$a_{ij} = i^{j-1}$，而且 $\boldsymbol{B} =$

$[b_{ij}]_{N\times 1}$，$b_{11}=N$，当 $i\geqslant 2$ 时 $b_{i1}=(i^N-1)/(i-1)$。对于 $N=3,7,11$ 的情况分别求解。精确解为 $X=[1\quad 1\ \cdots\ 1\quad 1]'$。对得到的结果与精确解的差异进行解释。

3. 修改程序 3.3，使得它可以通过重复求解 N 个线性方程组：

$$AC_J=E_J,\ J=1,2,\cdots,N$$

来得到 A^{-1}。

则：

$$A[C_1\quad C_2\quad \cdots\quad C_N]=[E_1\quad E_2\quad \cdots\quad E_N]$$

而且：

$$A^{-1}=[C_1\quad C_2\quad \cdots\quad C_N]$$

应确保对 LU 分解只计算一次！

4. 基尔霍夫电压定律说明电路网络中任意单向闭路的电压和为零。对图 3.5 中的电路利用基尔霍夫电压定律进行分析，可得到如下线性方程组：

$$\begin{aligned}(R_1+R_2+R_4)I_1+R_3I_2+R_4I_3&=E_1\\ R_3I_1+(R_2+R_3+R_5)I_2-R_5I_3&=E_2\\ R_4I_1-R_5I_2+(R_4+R_5+R_6)I_3&=0\end{aligned}\tag{17}$$

如果：

(a) $R_1=1,R_2=1,R_3=2,R_4=1,R_5=2,R_6=4$，而且 $E_1=23,E_2=29$

(b) $R_1=1,R_2=0.75,R_3=1,R_4=2,R_5=1,R_6=4$，而且 $E_1=12,E_2=21.5$

(c) $R_1=1,R_2=2,R_3=4,R_4=3,R_5=1,R_6=5$，而且 $E_1=41,E_2=38$

使用程序 3.3 求解电流 I_1,I_2,I_3。

图 3.5　练习 4 中的电路网络

5. 在微积分中，下列积分可通过部分分式技术求出：

$$\int\frac{x^2+x+1}{(x-1)(x-2)(x-3)^2(x^2+1)}dx$$

这需要在下列表达式中求出系数 $A_i,i=1,2,\cdots,6$：

$$\frac{x^2+x+1}{(x-1)(x-2)(x-3)^2(x^2+1)}=\frac{A_1}{(x-1)}+\frac{A_2}{(x-2)}+\frac{A_3}{(x-3)^2}+\frac{A_4}{(x-3)}+\frac{A_5x+A_6}{(x^2+1)}$$

使用程序 3.3 求解部分分式的系数。

6. 使用程序 3.3 求解线性方程组 $AX=B$，这里矩阵 A 由 MATLAB 命令 A = rand(10,10) 生成，B = [1 2 3 ⋯ 10]'。在使用程序 3.3 前，需要验证 A 是非奇异矩阵($\det(A)\neq 0$)。

通过计算 $\boldsymbol{AX}-\boldsymbol{B}$ 的差值来检查结果的精确性，并检查差值接近零的程度（一个精确解应使得 $\boldsymbol{AX}-\boldsymbol{B}=0$）。使用命令 A = rand(20,20)和 B = [1 2 3 … 20]'生成的矩阵 $\boldsymbol{A}$ 重复上述过程。解释用程序 3.3 求解这两个方程组在精确性上的明显区别。

7. 在 3.1 节的式(8)中，定义了 N 维空间中线性组合的概念。例如，向量(4, -3)等价于矩阵[4　-3]'，可表示为[1　0]'与[0　1]'的线性组合：

$$\begin{bmatrix} 4 \\ -3 \end{bmatrix} = 4\begin{bmatrix} 1 \\ 0 \end{bmatrix} + (-3)\begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

使用程序 3.3 来说明矩阵[1　3　5　7　9]'可表示为如下线性组合：

$$\begin{bmatrix} 0 \\ 4 \\ -2 \\ 3 \\ -1 \end{bmatrix}, \quad \begin{bmatrix} 2 \\ 0 \\ 0 \\ 4 \\ 4 \end{bmatrix}, \quad \begin{bmatrix} 3 \\ 2 \\ 0 \\ 5 \\ 1 \end{bmatrix}, \quad \begin{bmatrix} 5 \\ 6 \\ -3 \\ 0 \\ 2 \end{bmatrix}, \quad \begin{bmatrix} 1 \\ 4 \\ -2 \\ 7 \\ 0 \end{bmatrix}$$

解释为什么任意矩阵[x_1　x_2　x_3　x_4　x_5]'可表示为上述矩阵的线性组合。

3.6　求解线性方程组的迭代法

这一节主要讲述如何扩展第 2 章介绍的迭代法到更高维数。首先考虑用于线性方程组的固定点迭代的扩展。

3.6.1　雅克比迭代

例 3.26　考虑如下方程组：

$$\begin{aligned} 4x - y + z &= 7 \\ 4x - 8y + z &= -21 \\ -2x + y + 5z &= 15 \end{aligned} \tag{1}$$

上述方程可表示成如下形式：

$$\begin{aligned} x &= \frac{7+y-z}{4} \\ y &= \frac{21+4x+z}{8} \\ z &= \frac{15+2x-y}{5} \end{aligned} \tag{2}$$

这给出了下列雅克比迭代过程：

$$\begin{aligned} x_{k+1} &= \frac{7+y_k-z_k}{4} \\ y_{k+1} &= \frac{21+4x_k+z_k}{8} \\ z_{k+1} &= \frac{15+2x_k-y_k}{5} \end{aligned} \tag{3}$$

如果从 $\boldsymbol{P}_0=(x_0,y_0,z_0)=(1,2,2)$ 开始，则式(3)中的迭代将收敛到解(2,4,3)。

将 $x_0=1, y_0=2$ 和 $z_0=2$ 代入式(3)中每个方程的右边可得如下新值:

$$x_1=\frac{7+2-2}{4}=1.75$$

$$y_1=\frac{21+4+2}{8}=3.375$$

$$z_1=\frac{15+2-2}{5}=3.00$$

新的点 $\boldsymbol{P}_1=(1.75,3.375,3.00)$ 比 $\boldsymbol{P}_0$ 更接近(2,4,3)。使用式(3)的迭代过程生成点的序列 $\{\boldsymbol{P}_k\}$ 收敛到解(2,4,3)(如表 3.2 所示)。

表 3.2 求解线性方程组(1)的收敛的雅克比迭代

k	x_k	y_k	z_k
0	1.0	2.0	2.0
1	1.75	3.375	3.0
2	1.84375	3.875	3.025
3	1.9625	3.925	2.9625
4	1.99062500	3.97656250	3.00000000
5	1.99414063	3.99531250	3.00093750
⋮	⋮	⋮	⋮
15	1.99999993	3.99999985	2.99999993
⋮	⋮	⋮	⋮
19	2.00000000	4.00000000	3.00000000

这个过程称为雅克比迭代,可用来求解某些类型的线性方程组。经过 19 步迭代,迭代过程收敛到一个精度为 9 位有效数字的近似值(2.00000000,4.00000000,3.00000000)。

在求解偏微分方程中,线性方程组经常有多达 100 000 个变量。这些方程组的系数矩阵是稀疏矩阵,即系数矩阵中的大多数元素为零。如果非零元素具有一种模式(如三对角方程组),则迭代过程是求解这些大型方程组的有效方法。

有时雅克比迭代法是无效的。通过下面的例子可看出,重新排列初始线性方程组后,利用雅克比迭代法可产生一个发散的点的序列。

例 3.27 设重新排列线性方程组(1)如下:

$$\begin{aligned}-2x+y+5z&=15\\4x-8y+z&=-21\\4x-y+z&=7\end{aligned}\tag{4}$$

这些方程可表示为如下形式:

$$\begin{aligned}x&=\frac{-15+y+5z}{3}\\y&=\frac{21+4x+z}{8}\\z&=7-4x+y\end{aligned}\tag{5}$$

可用如下雅克比迭代过程求解:

$$x_{k+1}=\frac{-15+y_k+5z_k}{3}$$

$$y_{k+1} = \frac{21 + 4x_k + z_k}{8} \tag{6}$$

$$z_{k+1} = 7 - 4x_k + y_k$$

如果从 $P_0 = (x_0, y_0, z_0) = (1,2,2)$ 开始，则利用式(6)中的迭代将对解(2,4,3)发散。

将 $x_0 = 1, y_0 = 2$ 和 $z_0 = 2$ 代入式(6)中每个方程的右边可得到新值 x_1, y_1 和 z_1：

$$x_1 = \frac{-15 + 2 + 10}{2} = -1.5$$

$$y_1 = \frac{21 + 4 + 2}{8} = 3.375$$

$$z_1 = 7 - 4 + 2 = 5.00$$

新的点 $P_1 = (-1.5, 3.375, 5.00)$ 比 P_0 更远地偏离解(2,4,3)。利用式(6)中的迭代产生了一个发散序列(如表 3.3 所示)。

表 3.3　求解线性方程组(\.\.)的发散的雅克比迭代

k	x_k	y_k	z_k
0	1.0	2.0	2.0
1	-1.5	3.375	5.0
2	6.6875	2.5	16.375
3	34.6875	8.015625	-17.25
4	-46.617188	17.8125	-123.73438
5	-307.929688	-36.150391	211.28125
6	502.62793	-124.929688	1202.56836
⋮	⋮	⋮	⋮

3.6.2 Gauss-Seidel 迭代法

有时通过其他方法可使收敛速度加快。观察由雅克比迭代过程式(3)产生的 3 个序列 $\{x_k\}, \{y_k\}, \{z_k\}$，它们分别收敛到 2,4,3(如表 3.2 所示)。由于 x_{k+1} 被认为是比 x_k 更好的 x 的近似值，所以在计算 y_{k+1} 时，将 x_{k+1} 用来替换 x_k 是合理的。同理，可用 x_{k+1} 和 y_{k+1} 计算 z_{k+1}。下面的例子显示了对例 3.26 中的方程组使用上述方法的情况。

例 3.28　设给定式(1)中的线性方程组并利用 Gauss-Seidel 迭代过程求解：

$$x_{k+1} = \frac{7 + y_k - z_k}{4}$$

$$y_{k+1} = \frac{21 + 4x_{k+1} + z_k}{8} \tag{7}$$

$$z_{k+1} = \frac{15 + 2x_{k+1} - y_{k+1}}{5}$$

解：

如果从 $P_0 = (x_0, y_0, z_0) = (1,2,2)$ 开始，用式(7)中的迭代收敛到解(2,4,3)。

将 $y_0 = 2$ 和 $z_0 = 2$ 代入式(7)中第一个方程可得：

$$x_1 = \frac{7 + 2 - 2}{4} = 1.75$$

将 $x_1 = 1.75$ 和 $z_0 = 2$ 代入式(7)中第二个方程可得：

$$y_1 = \frac{21 + 4(1.75) + 2}{8} = 3.75$$

将 $x_1 = 1.75$ 和 $y_1 = 3.75$ 代入式(7)中第三个方程可得：

$$z_1 = \frac{15 + 2(1.75) - 3.75}{5} = 2.95$$

新的点 $\boldsymbol{P}_1 = (1.75, 3.75, 2.95)$ 比 $\boldsymbol{P}_0$ 更接近解(2,4,3)，而且比例 3.26 中的值更好。用式(7)的迭代生成序列 $\{\boldsymbol{P}_k\}$ 收敛到(2,4,3)(如表 3.4 所示)。

表 3.4 用于方程组(1)的收敛的 Gauss-Seidel 迭代

k	x_k	y_k	z_k
0	1.0	2.0	2.0
1	1.75	3.75	2.95
2	1.95	3.96875	2.98625
3	1.995625	3.99609375	2.99903125
⋮	⋮	⋮	⋮
8	1.99999983	3.99999988	2.99999996
9	1.99999998	3.99999999	3.00000000
10	2.00000000	4.00000000	3.00000000

在例 3.26 和例 3.27 中，有必要建立一些判定条件来判断雅克比迭代是否收敛。因此建立下面的定义。

定义 3.6 设有 N×N 阶矩阵 $\boldsymbol{A}$，如果：

$$|a_{kk}| > \sum_{\substack{j=1 \\ j \neq k}}^{N} |a_{kj}|, k = 1, 2, \cdots, N \tag{8}$$

则称 $\boldsymbol{A}$ 具有严格对角优势。

这表示在矩阵的每一行中，主对角线上元素的绝对值大于其他元素的绝对值的和。例 3.26 的线性方程组(1)的系数矩阵具有严格对角优势，原因在于：

在行 1 中：　$|4| > |-1| + |1|$

在行 2 中：　$|-8| > |4| + |1|$

在行 3 中：　$|5| > |-2| + |1|$

所有的行满足定义 3.6 中的关系式(8)，所以线性方程组(1)的系数矩阵 $\boldsymbol{A}$ 具有严格对角优势。

例 3.27 中的线性方程组(4)的系数矩阵 $\boldsymbol{A}$ 不具有严格对角占优，原因在于：

在行 1 中：　$|-2| < |1| + |5|$

在行 2 中：　$|-8| < |4| + |1|$

在行 3 中：　$|1| < |4| + |-1|$

行 1 和行 3 不满足定义 3.6 中的关系式(8)，因此线性方程组(4)中的系数矩阵 $\boldsymbol{A}$ 不具有严格对角优势。

现在生成雅克比迭代和 Gauss-Sediel 迭代过程。设有如下线性方程组：

$$\begin{array}{llllll}
a_{11}x_1+a_{12}x_2 & +\cdots+a_{1j}x_j+\cdots+ & a_{1N}x_N & =b_1 \\
a_{21}x_1+a_{22}x_2 & +\cdots+a_{2j}x_j+\cdots+ & a_{2N}x_N & =b_2 \\
\vdots \quad \vdots & \quad \vdots & \vdots & \vdots \\
a_{j1}x_1+a_{j2}x_2 & +\cdots+a_{jj}x_j+\cdots+ & a_{jN}x_N & =b_j \\
\vdots \quad \vdots & \quad \vdots & \vdots & \vdots \\
a_{N1}x_1+a_{N2}x_2 & +\cdots+a_{Nj}x_j+\cdots+ & a_{NN}x_N & =b_N
\end{array} \tag{9}$$

设第 k 点为 $P_k=(x_1^{(k)},x_2^{(k)},\cdots,x_j^{(k)},\cdots,x_N^{(k)})$，则下一点为 $P_{k+1}=(x_1^{(k+1)},x_2^{(k+1)},\cdots,x_j^{(k+1)},\cdots,x_N^{(k+1)})$。坐标 P_k 的上标(k)可用来标识属于这一点的坐标。迭代公式根据前面的值 $x_1^{(k)}$，$x_2^{(k)},\cdots,x_j^{(k)},\cdots,x_N^{(k)}$，使用式(9)中的行 j 求解式 $x_j^{(k+1)}$：

雅克比迭代：

$$x_j^{(k+1)}=\frac{b_j-a_{j1}x_1^{(k)}-\cdots-a_{jj-1}x_{j-1}^{(k)}-a_{jj+1}x_{j+1}^{(k)}-\cdots-a_{jN}x_N^{(k)}}{a_{jj}} \tag{10}$$

其中 $j=1,2,\cdots,N$

雅克比迭代使用所有老的坐标来生成所有新的坐标，而 Gauss-Seidel 迭代尽可能使用新的坐标得到更新的坐标：

Gauss-Seidel 迭代：

$$x_j^{(k+1)}=\frac{b_j-a_{j1}x_1^{(k+1)}-\cdots-a_{jj-1}x_{j-1}^{(k+1)}-a_{jj+1}x_{j+1}^{(k)}-\cdots-a_{jN}x_N^{(k)}}{a_{jj}} \tag{11}$$

其中 $j=1,2,\cdots,N$

下面的定理给出了雅克比迭代收敛的充分条件。

定理 3.15（雅克比迭代）　设矩阵 A 具有严格对角优势，则 $AX=B$ 有惟一解 $X=P$。利用式(10)的迭代可产生一个向量序列$\{P_k\}$，而且对于任意初始向量 P_0，向量序列都将收敛到 P。

证明：可参见有关数值分析的高级教材。

当矩阵 A 具有严格对角占优时，可证明 Gauss-Seidel 迭代法也会收敛。在大多数情况下，由于 Gauss-Seidel 迭代法比雅克比迭代法收敛得更快，所以，通常利用 Gauss-Seidel 迭代法(可比较例 3.26 和例 3.28)。为得到式(11)而对式(10)进行的修改是很重要的。在某些情况下，雅克比迭代会收敛而 Gauss-Seidel 迭代不会收敛。

3.6.3　收敛性

比较向量之间的距离是非常必要的，它可以用来判断$\{P_k\}$是否收敛到 P。$P=(x_1,x_2,\cdots,x_N)$和 $Q=(y_1,y_2,\cdots,y_N)$之间的欧几里德距离(参见 3.1 节)为：

$$\|P-Q\|=\left(\sum_{j=1}^{N}(x_j-y_j)^2\right)^{1/2} \tag{12}$$

它的缺点是需要相当大的计算量。因此引入另一种模 $\|X\|_1$：

$$\|X\|_1=\sum_{j=1}^{N}|x_j| \tag{13}$$

下面的结论保证了 $\|X\|_1$ 具有度量的数学结构，因此适合作为一个一般化的“距离公

式"。根据线性代数的理论可知,如果两个向量的 $\| * \|_1$ 模接近,则它们的欧几里德模 $\| * \|$ 也接近。

定理 3.16 设 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 是 N 维向量,c 是一个标量。则函数 $\|\boldsymbol{X}\|_1$ 有如下性质:

$$\|\boldsymbol{X}\|_1 \geq 0 \tag{14}$$

$$\|\boldsymbol{X}\|_1 = 0 \quad 当且仅当\ \boldsymbol{X} = 0 \tag{15}$$

$$\|c\boldsymbol{X}\|_1 = |c|\ \|\boldsymbol{X}\|_1 \tag{16}$$

$$\|\boldsymbol{X} + \boldsymbol{Y}\|_1 \leq \|\boldsymbol{X}\|_1 + \|\boldsymbol{Y}\|_1 \tag{17}$$

证明:这里只证明不等式(17),其他的留作练习。对于每个 j,实数的三角不等式表示为 $|x_j + y_j| \leq |x_j| + |y_j|$。根据这些不等式可得到不等式(17):

$$\|\boldsymbol{X} + \boldsymbol{Y}\|_1 = \sum_{j=1}^{N} |x_j + y_j| \leq \sum_{j=1}^{N} |x_j| + \sum_{j=1}^{N} |y_j| = \|\boldsymbol{X}\|_1 + \|\boldsymbol{Y}\|_1$$

可用式(13)定义的模来定义两点之间的距离。

定义 3.7 设 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 是 N 维空间中的两点。定义 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 的距离为 $\| * \|_1$ 模,表示为:

$$\|\boldsymbol{X} - \boldsymbol{Y}\|_1 = \sum_{j=1}^{N} |x_j - y_j|$$

例 3.29 计算点 $\boldsymbol{P} = (2,4,3)$ 和 $\boldsymbol{Q} = (1.75,3.75,2.95)$ 的欧几里德距离和 $\| * \|_1$ 距离。

解:

欧几里德距离为:

$$\|\boldsymbol{P} - \boldsymbol{Q}\| = ((2-1.75)^2 + (4-3.75)^2 + (3-2.95)^2)^{1/2} = 0.3570$$

$\| * \|_1$ 距离为:

$$\|\boldsymbol{P} - \boldsymbol{Q}\|_1 = |2-1.75| + |4-3.75| + |3-295| = 0.55$$

$\| * \|_1$ 更容易计算,常用来确定 N 维空间中的收敛性。

在程序 3.4 中使用了 MATLAB 命令 A(j,[1:j-1,j+1:N])。它有效地选择 A 中行 j 的所有元素,但不包括位于第 j 列的元素(即 A(j,j))。这种表示可简化程序 3.4 中的雅克比迭代步骤。

在程序 3.4 和程序 3.5 中,使用了 MATLAB 命令 norm,它是欧几里德模。也可以使用 $\| * \|_1$。详细内容请查阅 MATLAB 中与 norm 命令相关的帮助信息和参考信息。

程序 3.4(雅克比迭代) 求解线性方程组 $\boldsymbol{AX} = \boldsymbol{B}$。初始值 $\boldsymbol{X} = \boldsymbol{P}_0$,并生成序列 $\{\boldsymbol{P}_k\}$,最后收敛到解。程序可用的充分条件是 A 具有严格对角优势

```
function X = jacobi(A,B,P,delta,max1)
% Input  - A is an N x N nonsingular matrix
%        - B is an N x 1 matrix
%        - P is an N x 1 matrix;the initial guess
%        - delta is the tolerance for P
%        - max1 is the maximum number of iterations
% Output - X is an N x 1 matrix:the jacobi approximation to
%          the solution of AX = B
N = length(B);

for k = 1:max1
```

```
   for j=1:N
      X(j)=(B(j)-A(j,[1:j-1,j+1:N])*P([1:j-1,j+1:N]))/A(j,j);
   end
   err=abs(norm(X'-P));
   relerr=err/(norm(X)+eps);
   P=X';
      if(err<delta)|(relerr<delta)
      break
   end
end
X=X';
```

程序 3.5（Gauss-Seidel 迭代） 求解线性方程组 $\boldsymbol{AX}=\boldsymbol{B}$。初始值 $\boldsymbol{X}=\boldsymbol{P}_0$，并生成序列 $\{\boldsymbol{P}_k\}$，最后收敛到解。程序可用的充分条件是 $\boldsymbol{A}$ 具有严格对角优势

```
function X=gseid(A,B,P,delta,max1)
% Input  - A is an N x N nonsingular matrix
%        - B is an N x 1 matrix
%        - P is an N x 1 matrix;the initial guess
%        - delta is the tolerance for P
%        - max1 is the maximum number of iterations
% Output - X is an N x 1 matrix:the gauss-seidel
%          approximation to the solution of AX=B
N = length(B);
for k=1:max1
   for j=1:N
      if j==1
         X(1)=(B(1)-A(1,2:N)*P(2:N))/A(1,1);
      elseif j==N
         X(N)=(B(N)-A(N,1:N-1)*(X(1:N-1))')/A(N,N);
      else
         %X contains the kth approximations and P the (k-1)st
         X(j)=(B(j)-A(j,1:j-1)*X(1:j-1)
               -A(j,j+1:N)*P(j+1:N))/A(j,j);
      end
   end
   err=abs(norm(X'-P));
   relerr=err/(norm(X)+eps);
   P=X';
      if (err<delta)|(relerr<delta)
         break
      end
end
X=X';
```

3.6.4 求解线性方程组的迭代法的练习

在练习 1 到练习 8 中：

(a) 初始值 $\boldsymbol{P}_0=0$，利用雅克比迭代求解 $\boldsymbol{P}_k,k=1,2,3$。雅克比迭代收敛到解吗？

(b) 初始值 $\boldsymbol{P}_0=0$，利用 Gauss-Seidel 迭代求解 $\boldsymbol{P}_k,k=1,2,3$。Gauss-Seidel 迭代收敛到解吗？

1. $4x-\ y=\ 15$　　　　2. $8x-3y=\ 10$

$x+5y=9$ $\quad$ $-x+4y=6$

3. $\begin{aligned} -x+3y&=1\\ 6x-2y&=2\end{aligned}$
4. $\begin{aligned} 2x+3y&=1\\ 7x-2y&=1\end{aligned}$
5. $\begin{aligned} 5x-y+z&=10\\ 2x+8y-z&=11\\ -x+y+4z&=3\end{aligned}$
6. $\begin{aligned} 2x+8y-z&=11\\ 5x-y+z&=10\\ -x+y+4z&=3\end{aligned}$
7. $\begin{aligned} x-5y-z&=-8\\ 4x+y-z&=13\\ 2x-y-6z&=-2\end{aligned}$
8. $\begin{aligned} 4x+y-z&=13\\ x-5y-z&=-8\\ 2x-y-6z&=-2\end{aligned}$
9. 设 $\boldsymbol{X}=(x_1,x_2,\cdots,x_N)$。证明 $\|*\|_1$ 模:

$$\|\boldsymbol{X}\|_1=\sum_{k=1}^{N}|x_k|$$

满足性质(14)到性质(16)。

10. 设 $\boldsymbol{X}=(x_1,x_2,...,x_N)$。证明欧几里德模:

$$\|\boldsymbol{X}\|=\left(\sum_{k=1}^{N}(x_k)^2\right)^{1/2}$$

满足性质(14)到性质(17)。

11. 设 $\boldsymbol{X}=(x_1,x_2,\cdots,x_N)$。证明 $\|*\|_\infty$ 模:

$$\|\boldsymbol{X}\|_\infty=\max_{1\leqslant k\leqslant N}|x_k|$$

满足性质(14)到性质(17)。

3.6.5 算法和程序

1. 使用程序3.4和程序3.5求解练习1到练习8中的线性方程组。使用format long命令和delta $=10^{-9}$。
2. 在定理3.14中,$\boldsymbol{A}$ 具有严格对角优势是充分条件,不是必要条件。使用程序3.4,程序3.5以及多个不同的初始值 $\boldsymbol{P}_0$ 对下列线性方程组求解(注意:可能雅克比迭代收敛,但Gauss-Seidel迭代发散):

$$\begin{aligned} x\qquad\;+z&=2\\ -x+y\qquad&=0\\ x+2y-3z&=0\end{aligned}$$

3. 设有如下三角线性方程组,而且系数矩阵具有严格对角优势:

$$\begin{aligned}
d_1x_1+c_1x_2\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad&=b_1\\
a_1x_1+d_2x_2+c_2x_3\qquad\qquad\qquad\qquad\qquad\qquad&=b_2\\
a_2x_2+d_3x_3+c_3x_4\qquad\qquad\qquad\qquad&=b_3\\
\vdots\qquad\qquad\qquad\qquad&\quad\vdots\\
a_{N-2}x_{N-2}+d_{N-1}x_{N-1}+c_{N-1}x_N&=b_{N-1}\\
a_{N-1}x_{N-1}+d_Nx_N&=b_N
\end{aligned}$$

(i) 根据式(9)到式(11),设计一算法来求解上述方程组。算法必须有效地利用系数矩

阵的稀疏性。

(ii) 根据(i)中设计的算法构造一个 MATLAB 程序,并求解下列三角线性方程组:

(a)
$$\begin{aligned}
4m_1 + m_2 &= 3\\
m_1 + 4m_2 + m_3 &= 3\\
m_2 + 4m_3 + m_4 &= 3\\
m_3 + 4m_4 + m_5 &= 3\\
\vdots \quad \vdots \quad \vdots \quad &\vdots\\
m_{48} + 4m_{49} + m_{50} &= 3\\
m_{49} + 4m_{50} &= 3
\end{aligned}$$

(b)
$$\begin{aligned}
4m_1 + m_2 &= 1\\
m_1 + 4m_2 + m_3 &= 2\\
m_2 + 4m_3 + m_4 &= 1\\
m_3 + 4m_4 + m_5 &= 2\\
\vdots \quad \vdots \quad \vdots \quad &\vdots\\
m_{48} + 4m_{49} + m_{50} &= 1\\
m_{49} + 4m_{50} &= 2
\end{aligned}$$

4. 利用 Gauss-Seidel 迭代法求解下列带状方程:

$$\begin{aligned}
12x_1 - 2x_2 + x_3 &= 5\\
-2x_1 + 12x_2 - 2x_3 + x_4 &= 5\\
x_1 - 2x_2 + 12x_3 - 2x_4 + x_5 &= 5\\
x_2 - 2x_3 + 12x_4 - 2x_5 + x_6 &= 5\\
\vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \quad &\vdots\\
x_{46} - 2x_{47} + 12x_{48} - 2x_{49} + x_{50} &= 5\\
x_{47} - 2x_{48} + 12x_{49} - 2x_{50} &= 5\\
x_{48} - 2x_{49} + 12x_{50} &= 5
\end{aligned}$$

5. 在程序 3.4 和程序 3.5 中,将相邻迭代之间的相对误差作为终止评定条件。在 2.3 节中已讨论了独立使用这个判定条件的问题。线性方程组 $\boldsymbol{AX}=\boldsymbol{B}$ 可重写为 $\boldsymbol{AX}-\boldsymbol{B}=\boldsymbol{0}$。如果 $\boldsymbol{X}_k$ 是雅克比迭代或 Gauss-Seidel 迭代过程中的第 k 个迭代值,则一般情况下,余项 $\boldsymbol{AX}_k-\boldsymbol{B}$ 的模是一个更适合的中止评定条件。

修改程序 3.4 和程序 3.5,使用余项的模作为中止评定条件。使用修改后的程序求解问题 4 中的带状方程组。

3.7 非线性方程组的迭代法:Seidel 法和牛顿法(可选)

本节讨论如何扩展第 2 章和 3.6 节的迭代法以求解非线性方程组。考虑下列函数:

$$\begin{aligned}
f_1(x,y) &= x^2-2x-y+0.5\\
f_2(x,y) &= x^2+4y^2-4
\end{aligned} \tag{1}$$

需要寻找一个方法求解非线性函数构成的方程组:

$$f_1(x,y)=0 \quad 和 \quad f_2(x,y)=0 \tag{2}$$

方程 $f_1(x,y)=0$ 和 $f_2(x,y)=0$ 隐含定义了在 xy 平面上的曲线。因此方程组(2)的解是两个曲线都经过一个点(即 $f_1(p,q)=0$ 且 $f_2(p,q)=0$)。方程组(1)表示的曲线如下:

$$\begin{aligned}
x^2-2x-y+0.5=0 &\quad 是抛物线\\
x^2+4y^2-4=0 &\quad 是椭圆
\end{aligned} \tag{3}$$

图 3.6 中的图形显示存在两个解,在(−0.2,1.0)和(1.9,0.3)附近。

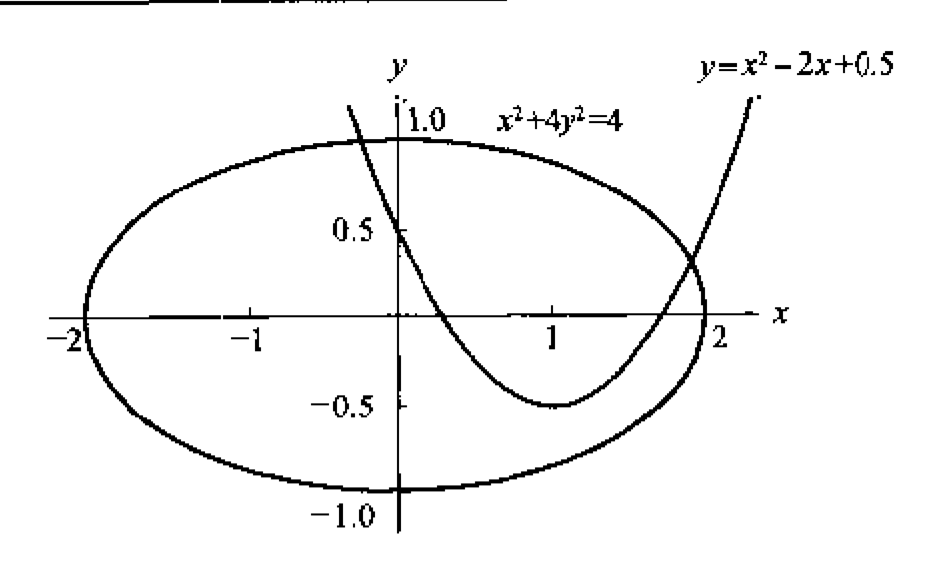


图 3.6 非线性函数 $y=x^2-2x+0.5$ 和 $x^2+4y^2=4$ 的图形

第一种技术是固定点迭代法。必须设计一个方法来生成序列$\{(p_k,q_k)\}$,使其收敛到解(p,q)。式(3)中的第一个方程可用来求解 x。而 y 的一个倍数要加到第二个方程的两边,得到 $x^2+4y^2-8y-4=-8y$。选择增加 $-8y$ 很重要,原因将在以后解释。现在可得到一个等价的方程组:

$$x=\frac{x^2-y+0.5}{2}$$
$$y=\frac{-x^2-4y^2+8y+4}{8} \tag{4}$$

这两个方程可用来构造递归公式。设初始点为(p_0,q_0),可利用下列递归公式计算序列$\{(p_{k+1},q_{k+1})\}$:

$$p_{k+1}=g_1(p_k,q_k)=\frac{p_k^2-q_k+0.5}{2}$$
$$q_{k+1}=g_2(p_k,q_k)=\frac{-p_k^2-4q_k^2+8q_k+4}{8} \tag{5}$$

情况(i) 设初始值$(p_0,q_0)=(0,1)$,则:

$$p_1=\frac{0^2-1+0.5}{2}=-0.25 \quad 和 \quad q_1=\frac{-0^2-4(1)^2+8(1)+4}{8}=1.0$$

迭代过程将生成表 3.5 中情况(i)下面的序列。在这种情况下,序列收敛到(0,1)附近的解。

表 3.5 利用(5)中公式的固定点迭代

情况(i):从(0,1)开始			情况(ii):从(0,2)开始		
k	p_k	q_k	k	p_k	q_k
0	0.00	1.00	0	2.00	0.00
1	-0.25	1.00	1	2.25	0.00
2	-0.21875	0.9921875	2	2.78125	-0.1328125
3	-0.2221680	0.99398880	3	4.184082	-0.6085510
4	-0.2223147	0.9938121	4	9.307547	-2.4820360
5	-0.2221941	0.9938029	5	44.80623	-15.891091
6	-0.2222163	0.9938095	6	1 011.995	-392.60426
7	-0.2222147	0.9938083	7	512 263.2	-205 477.82
8	-0.2222145	0.9938084	序列发散		
9	-0.2222146	0.9938084			

情况(ii) 设初始值$(p_0,q_0)=(2,0)$,则:

$$p_1=\frac{2^2-0+0.5}{2}=2.25 \quad 和 \quad q_1=\frac{-2^2-4(0)^2+8(0)+4}{8}=0.0$$

迭代过程将生成表 3.5 中情况(ii)下的序列。在这种情况下,序列是发散的。

利用式(5)的迭代过程不能找到第二个解(1.900677,0.3112186)。为了找到这个点,需要另一对不同的迭代公式。在式(3)中,将第一个方程加 $-2x$,第二个方程加 $-11y$,表示为:

$$x^2-4x-y+0.5=-2x \quad 和 \quad x^2+4y^2-11y-4=-11y$$

通过上述方程可得到迭代公式:

$$\begin{aligned} p_{k+1}&=g_1(p_k,q_k)=\frac{-p_k^2+4p_k+q_k-0.5}{2}\\ q_{k+1}&=g_2(p_k,q_k)=\frac{-p_k^2-4q_k^2+11q_k+4}{11} \end{aligned} \tag{6}$$

表 3.6 显示了如何利用式(6)求第二个解。

表 3.6　使用式(6)中的固定点迭代

k	p_k	q_k
0	2.00	0.00
1	1.75	0.0
2	1.71875	0.0852273
3	1.753063	0.1776676
4	1.808345	0.2504410
8	1.903595	0.3160782
12	1.900924	0.3112267
16	1.900652	0.3111994
20	1.900677	0.3112196
24	1.900677	0.3112186

3.7.1 理论

为何式(6)中的公式适合求(1.9,0.3)附近的解,而式(5)中的公式不适合?在 2.1 节中,通过分析固定点导数的大小可得出答案。当存在多个变量的函数时,必须使用偏导。将用于存在多个变量的函数的方程组的一般"导数"定义为雅克比矩阵(又称导数矩阵)。这里只介绍一些基本的思想,更多的细节可参见高等微积分教材。

定义 3.8(雅克比矩阵)　设 $f_1(x,y)$ 和 $f_2(x,y)$ 是包含自变量 x 和 y 的函数,则它们的雅克比矩阵 $\boldsymbol{J}(x,y)$ 表示为:

$$\begin{bmatrix} \dfrac{\partial f_1}{\partial x} & \dfrac{\partial f_1}{\partial y} \\ \dfrac{\partial f_2}{\partial x} & \dfrac{\partial f_2}{\partial y} \end{bmatrix} \tag{7}$$

同理,如果 $f_1(x,y,z)$,$f_2(x,y,z)$,$f_3(x,y,z)$ 是包含自变量 x,y,z 的函数,则 3×3 雅克比矩阵 $\boldsymbol{J}(x,y,z)$ 定义为:

$$\begin{bmatrix} \frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} & \frac{\partial f_1}{\partial z} \\ \frac{\partial f_2}{\partial x} & \frac{\partial f_2}{\partial y} & \frac{\partial f_2}{\partial z} \\ \frac{\partial f_3}{\partial x} & \frac{\partial f_3}{\partial y} & \frac{\partial f_3}{\partial z} \end{bmatrix} \tag{8}$$

例 3.30 对下列 3 个函数求解在点(1,3,2)处的 3×3 雅克比矩阵 $\boldsymbol{J}(x,y,z)$:

$$f_1(x,y,z)=x^3-y^2+y-z^4+z^2$$
$$f_2(x,y,z)=xy+yz+xz$$
$$f_3(x,y,z)=\frac{y}{xz}$$

解:

雅克比矩阵为:

$$\boldsymbol{J}(x,y,z)=\begin{bmatrix} \frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} & \frac{\partial f_1}{\partial z} \\ \frac{\partial f_2}{\partial x} & \frac{\partial f_2}{\partial y} & \frac{\partial f_2}{\partial z} \\ \frac{\partial f_3}{\partial x} & \frac{\partial f_3}{\partial y} & \frac{\partial f_3}{\partial z} \end{bmatrix}=\begin{bmatrix} 3x^2 & -2y+1 & -4z^3+2z \\ y+z & x+z & y+x \\ \frac{-y}{x^2z} & \frac{1}{xz} & \frac{-y}{xz^2} \end{bmatrix}$$

这样,在点(1,3,2)处的雅克比矩阵为 3×3 矩阵:

$$\boldsymbol{J}(1,3,2)=\begin{bmatrix} 3 & -5 & -28 \\ 5 & 3 & 4 \\ -\frac{3}{2} & \frac{1}{2} & -\frac{3}{4} \end{bmatrix}$$

3.7.2 广义微分

对于含多个变量的函数,使用微分来表示自变量的变化情况如何影响因变量。设有如下表达式:

$$u=f_1(x,y,z),v=f_2(x,y,z),\omega=f_3(x,y,z) \tag{9}$$

设已知式(9)中函数在点(x_0,y_0,z_0)处的值,现在希望可以预测在邻近点(x,y,z)处的值。设 du,dv,dw 表示因变量的微分变化,而 dx,dy,dz 表示自变量的微分变化。这些变化服从如下关系:

$$\begin{aligned} du&=\frac{\partial f_1}{\partial x}(x_0,y_0,z_0)dx+\frac{\partial f_1}{\partial y}(x_0,y_0,z_0)dy+\frac{\partial f_1}{\partial z}(x_0,y_0,z_0)dz \\ dv&=\frac{\partial f_2}{\partial x}(x_0,y_0,z_0)dx+\frac{\partial f_2}{\partial y}(x_0,y_0,z_0)dy+\frac{\partial f_2}{\partial z}(x_0,y_0,z_0)dz \\ dw&=\frac{\partial f_3}{\partial x}(x_0,y_0,z_0)dx+\frac{\partial f_3}{\partial y}(x_0,y_0,z_0)dy+\frac{\partial f_3}{\partial z}(x_0,y_0,z_0)dz \end{aligned} \tag{10}$$

如果使用向量表示,则式(10)可通过使用雅克比矩阵进行简化。函数的变化用 $d\boldsymbol{F}$ 表示,变量的变化用 $d\boldsymbol{X}$ 表示:

$$dF=\begin{bmatrix} du \\ dv \\ d\omega \end{bmatrix}=J(x_0,y_0,z_0)\begin{bmatrix} dx \\ dy \\ dz \end{bmatrix}=J(x_0,y_0,z_0)dX \tag{11}$$

例 3.31　对如下方程，当自变量从(1,3,2)变化到(1.02,2.97,2.01)时，使用雅克比矩阵求微分变化$(du,dv,d\omega)$：

$$u=f_1(x,y,z)=x^3-y^2+y-z^4+z^2$$
$$v=f_2(x,y,z)=xy+yz+xz$$
$$\omega=f_3(x,y,z)=\frac{y}{xz}$$

解：

利用表达式(11)和例 3.30 的 $J(1,3,2)$，以及微分变化$(dx,dy,dz)=(0.02,-0.03,0.01)$可得到：

$$\begin{bmatrix} du \\ dv \\ d\omega \end{bmatrix}=\begin{bmatrix} 3 & -5 & -28 \\ 5 & 3 & 4 \\ -\frac{3}{2} & \frac{1}{2} & -\frac{3}{4} \end{bmatrix}\begin{bmatrix} 0.02 \\ -0.03 \\ 0.01 \end{bmatrix}=\begin{bmatrix} -0.07 \\ 0.05 \\ -0.0525 \end{bmatrix}$$

注意在点(1.02,2.97,2.01)处的函数值接近方程的近似解，即微分值 $du=-0.07$，$dv=0.05$，$d\omega=-0.0525$ 加上对应的函数值 $f_1(1,3,2)=-17$，$f_2(1,3,2)=11$，$f_3(1,3,2)=1.5$，表示为：

$$f_1(1.02,2.97,2.01)=-17.072\approx-17.01=f_1(1,3,2)+du$$
$$f_2(1.02,2.97,2.01)=11.0493\approx11.05=f_2(1,3,2)+dv$$
$$f_3=(1.02,2.97,2.01)=1.44864\approx1.4475=f_3(1,3,2)+d\omega$$

3.7.3　接近固定点处的收敛性

现在针对二维和三维函数给出 2.1 节中定义和定理的推广，这里没有使用 N 维函数的表示。读者可在许多有关数值分析的教材中找到这些推广。

定义 3.9　包含两个方程：

$$x=g_1(x,y)\quad 和\quad y=g_2(x,y) \tag{12}$$

的方程组的固定点是点(p,q)，满足 $p=g_1(p,q)$且 $q=g_2(p,q)$。在三维情况下，方程组：

$$x=g_1(x,y,z),y=g_2(x,y,z),z=g_3(x,y,z) \tag{13}$$

的固定点是点(p,q,r)，满足 $p=g_1(p,q,r)$，$q=g_2(p,q,r)$且 $r=g_3(p,q,r)$。

定义 3.10　对于(12)中的函数，固定点迭代为：

$$p_{k+1}=g_1(p_k,q_k)\quad 和\quad q_{k+1}=g_2(p_k,q_k) \tag{14}$$

其中 $k=0,1,\cdots$。同理，对(13)中的函数，固定点迭代为：

$$\begin{aligned} p_{k+1}&=g_1(p_k,q_k,r_k) \\ q_{k+1}&=g_2(p_k,q_k,r_k) \\ r_{k+1}&=g_3(p_k,q_k,r_k) \end{aligned} \tag{15}$$

其中 $k=0,1,\cdots$。

定理 3.17(固定点迭代) 设式(12)和式(13)中的函数及它们的一阶偏导数分别在包含(p,q)或(p,q,r)的区域内连续。如果初始点值足够接近固定点,则有下面两种情况。

情况(i) 二维情况:如果(p_0,q_0)足够接近(p,q),而且:

$$\left|\frac{\partial g_1}{\partial x}(p,q)\right|+\left|\frac{\partial g_1}{\partial y}(p,q)\right|<1$$
$$\left|\frac{\partial g_2}{\partial x}(p,q)\right|+\left|\frac{\partial g_2}{\partial y}(p,q)\right|<1 \tag{16}$$

则式(14)中的迭代将收敛到固定点(p,q)。

情况(ii) 三维情况:如果(p_0,q_0,r_0)足够接近(p,q,r),而且:

$$\left|\frac{\partial g_1}{\partial x}(p,q,r)\right|+\left|\frac{\partial g_1}{\partial y}(p,q,r)\right|+\left|\frac{\partial g_1}{\partial z}(p,q,r)\right|<1$$
$$\left|\frac{\partial g_2}{\partial x}(p,q,r)\right|+\left|\frac{\partial g_2}{\partial y}(p,q,r)\right|+\left|\frac{\partial g_2}{\partial z}(p,q,r)\right|<1$$
$$\left|\frac{\partial g_3}{\partial x}(p,q,r)\right|+\left|\frac{\partial g_3}{\partial y}(p,q,r)\right|+\left|\frac{\partial g_3}{\partial z}(p,q,r)\right|<1 \tag{17}$$

则式(15)中的迭代将收敛到固定点(p,q,r)。

如果条件(16)或条件(17)不满足,则迭代可能发散。这种情况通常发生在偏导绝对值之和远远大于1时。利用定理3.17可说明为何迭代式(5)可收敛到固定点$(-0.2,1.0)$。计算函数g_1和g_2的偏导可得:

$$\frac{\partial}{\partial x}g_1(x,y)=x,\qquad \frac{\partial}{\partial y}g_1(x,y)=-\frac{1}{2}$$
$$\frac{\partial}{\partial x}g_2(x,y)=-\frac{x}{4},\qquad \frac{\partial}{\partial y}g_2(x,y)=-y+1$$

实际上,对所有的(x,y),如果满足$-0.5<x<0.5$和$0.5<y<1.5$,则函数g_1和g_2的偏导满足:

$$\left|\frac{\partial}{\partial x}g_1(x,y)\right|+\left|\frac{\partial}{\partial y}g_1(x,y)\right|=|x|+|-0.5|<1$$
$$\left|\frac{\partial}{\partial x}g_2(x,y)\right|+\left|\frac{\partial}{\partial y}g_2(x,y)\right|=\frac{|-x|}{4}+|-y+1|<0.625<1$$

因此,式(16)中的偏导条件满足,而且根据定理3.17,固定点迭代将收敛到$(p,q)\approx(-0.2222146,0.9938084)$。在其他固定点$(1.00068,0.31122)$附近,偏导不满足式(16)中的条件,所以收敛性得不到保证。即:

$$\left|\frac{\partial}{\partial x}g_1(1.90068,0.31122)\right|+\left|\frac{\partial}{\partial y}g_1(1.90068,0.31122)\right|=2.40068>1$$
$$\left|\frac{\partial}{\partial x}g_2(1.90068,0.31122)\right|+\left|\frac{\partial}{\partial y}g_2(1.90068,0.31122)\right|=1.16395>1$$

3.7.4 Seidel 迭代

现在可构造一个与 Gauss-Seidel 法类似的改进型固定点迭代法。设用p_{k+1}计算q_{k+1}(在三维情况下,用p_{k+1}和q_{k+1}计算r_{k+1}),并将这些改进融入式(14)和式(15)中时,这个方法称为

Seidel 迭代：

$$p_{k+1} = g_1(p_k, q_k) \quad 和 \quad q_{k+1} = g_2(p_{k+1}, q_k) \tag{18}$$

以及：

$$\begin{aligned} p_{k+1} &= g_1(p_k, q_k, r_k) \\ q_{k+1} &= g_2(p_{k+1}, q_k, r_k) \\ r_{k+1} &= g_3(p_{k+1}, q_{k+1}, r_k) \end{aligned} \tag{19}$$

程序 3.6 实现了求解非线性方程组的 Seidel 迭代。固定点迭代的实现留给读者完成。

3.7.5 求解非线性方程组的牛顿法

现在将牛顿法推广到二维情况，也可以很容易地推广到更高维。

设有方程组：

$$\begin{aligned} u &= f_1(x, y) \\ v &= f_2(x, y) \end{aligned} \tag{20}$$

它意味着从 xy 平面到 uv 平面的变换。这里只关心在点 (x_0, y_0) 处的变换行为，即点 (u_0, v_0)。如果两个函数有连续的偏导，则在点 (x_0, y_0) 处用微分表示下列线性近似方程组是合法的：

$$\begin{aligned} u - u_0 &\approx \frac{\partial}{\partial x} f_1(x_0, y_0)(x - x_0) + \frac{\partial}{\partial y} f_1(x_0, y_0)(y - y_0) \\ v - v_0 &\approx \frac{\partial}{\partial x} f_2(x_0, y_0)(x - x_0) + \frac{\partial}{\partial y} f_2(x_0, y_0)(y - y_0) \end{aligned} \tag{21}$$

方程组(21)是一个局部线性变换，它将自变量的微小变化与因变量的微小变化联系起来。当使用雅克比矩阵 $\boldsymbol{J}(x_0, y_0)$ 时，这个关系可更容易地表示为：

$$\begin{bmatrix} u - u_0 \\ v - v_0 \end{bmatrix} = \begin{bmatrix} \frac{\partial}{\partial x} f_1(x_0, y_0) & \frac{\partial}{\partial y} f_1(x_0, y_0) \\ \frac{\partial}{\partial x} f_2(x_0, y_0) & \frac{\partial}{\partial y} f_2(x_0, y_0) \end{bmatrix} \begin{bmatrix} x - x_0 \\ y - y_0 \end{bmatrix} \tag{22}$$

如果式(20)中的方程组用向量函数 $\boldsymbol{V} = \boldsymbol{F}(\boldsymbol{X})$ 表示，则雅克比矩阵 $\boldsymbol{J}(x, y)$ 是导数的二维近似，因为式(22)可表示为：

$$\Delta \boldsymbol{F} \approx \boldsymbol{J}(x_0, y_0) \Delta \boldsymbol{X} \tag{23}$$

现在可以利用式(23)推导二维情况下的牛顿法。

设方程组(20)中，u 和 v 为 0：

$$\begin{aligned} 0 &= f_1(x, y) \\ 0 &= f_2(x, y) \end{aligned} \tag{24}$$

设 (p, q) 为式(24)的一个解，即：

$$\begin{aligned} 0 &= f_1(p, q) \\ 0 &= f_2(p, q) \end{aligned} \tag{25}$$

为了使用牛顿法求解式(24)，需要考虑函数在点 (p_0, q_0) 处的微小变化：

$$\begin{aligned} \Delta u &= u - u_0, \quad \Delta p = x - p_0 \\ \Delta v &= v - v_0, \quad \Delta q = y - q_0 \end{aligned} \tag{26}$$

设式(20)中 $(x, y) = (p, q)$，并利用式(25)，可得到 $(u, v) = (0,0)$。因此因变量的变化是：

$$u - u_0 = f_1(p,q) - f_1(p_0,q_0) = 0 - f_1(p_0,q_0) \tag{27}$$
$$v - v_0 = f_2(p,q) - f_2(p_0,q_0) = 0 - f_2(p_0,q_0)$$

将式(27)中的结果代入式(22)可得线性变换表达式:

$$\begin{bmatrix} \frac{\partial}{\partial x}f_1(p_0,q_0) & \frac{\partial}{\partial y}f_1(p_0,q_0) \\ \frac{\partial}{\partial x}f_2(p_0,q_0) & \frac{\partial}{\partial y}f_2(p_0,q_0) \end{bmatrix} \begin{bmatrix} \Delta p \\ \Delta q \end{bmatrix} \approx - \begin{bmatrix} f_1(p_0,q_0) \\ f_2(p_0,q_0) \end{bmatrix} \tag{28}$$

如果(28)中的雅克比矩阵 $\boldsymbol{J}(p_0,q_0)$非奇异,则可解出 $\Delta\boldsymbol{P}[\Delta p \quad \Delta q]' = [p \quad q]' - [p_0 \quad q_0]'$为:

$$\Delta\boldsymbol{P} \approx -\boldsymbol{J}(p_0,q_0)^{-1}\boldsymbol{F}(p_0,q_0) \tag{29}$$

然后,解 $\boldsymbol{P}_1$ 的下一个近似值 $\boldsymbol{P} = [p,q]'$ 为:

$$\boldsymbol{P}_1 = \boldsymbol{P}_0 + \Delta\boldsymbol{P} = \boldsymbol{P}_0 - \boldsymbol{J}(p_0,q_0)^{-1}\boldsymbol{F}(p_0,q_0) \tag{30}$$

注意式(30)是用于一个变量的牛顿法的一般化,即 $p_1 = p_0 - f(p_0)/f'(p_0)$。

3.7.6 牛顿法概要

设 P_k 已知。

步骤 1 计算函数:

$$\boldsymbol{F}(\boldsymbol{P}_k) = \begin{bmatrix} f_1(p_k,q_k) \\ f_2(p_k,q_k) \end{bmatrix}$$

步骤 2 计算雅克比矩阵:

$$\boldsymbol{J}(\boldsymbol{P}_k) = \begin{bmatrix} \frac{\partial}{\partial x}f_1(p_k,q_k) & \frac{\partial}{\partial y}f_1(p_k,q_k) \\ \frac{\partial}{\partial x}f_2(p_k,q_k) & \frac{\partial}{\partial y}f_2(p_k,q_k) \end{bmatrix}$$

步骤 3 求线性方程组:

$$\boldsymbol{J}(\boldsymbol{P}_k)\Delta\boldsymbol{P} = -\boldsymbol{F}(\boldsymbol{P}_k) \quad \text{中 } \Delta\boldsymbol{P} \text{ 的解}$$

步骤 4 计算下一点:

$$\boldsymbol{P}_{k+1} = \boldsymbol{P}_k + \Delta\boldsymbol{P}$$

重复上述过程。

例 3.32 设有非线性方程组:

$$0 = x^2 - 2x - y + 0.5$$
$$0 = x^2 + 4y^2 - 4$$

设初始值$(p_0,q_0) = (2,000,0.25)$,用牛顿法计算(p_1,q_1),(p_2,q_2),(p_3,q_3)。

解:

函数向量和雅克比矩阵为:

$$\boldsymbol{F}(x,y) = \begin{bmatrix} x^2 - 2x - y + 0.5 \\ x^2 + 4y^2 - 4 \end{bmatrix}, \quad \boldsymbol{J}(x,y) = \begin{bmatrix} 2x - 2 & -1 \\ 2x & 8y \end{bmatrix}$$

在点(2.00,0.25)处的值为:

$$\boldsymbol{F}(2.00,0.25)=\begin{bmatrix}0.25\\0.25\end{bmatrix},\quad \boldsymbol{J}(2.00,0.25)=\begin{bmatrix}2.0 & -1.0\\4.0 & 2.0\end{bmatrix}$$

微分 Δp 和 Δq 是下列线性方程组的解：

$$\begin{bmatrix}2.0 & -1.0\\4.0 & 2.0\end{bmatrix}\begin{bmatrix}\Delta p\\\Delta q\end{bmatrix}=-\begin{bmatrix}0.25\\0.25\end{bmatrix}$$

通过直接计算可得：

$$\Delta\boldsymbol{P}=\begin{bmatrix}\Delta p\\\Delta q\end{bmatrix}=-\begin{bmatrix}-0.09375\\0.0625\end{bmatrix}$$

迭代的下一点为：

$$\boldsymbol{P}_1=\boldsymbol{P}_0+\Delta\boldsymbol{P}=\begin{bmatrix}2.00\\0.25\end{bmatrix}+\begin{bmatrix}-0.09375\\0.0625\end{bmatrix}=\begin{bmatrix}1.90625\\0.3125\end{bmatrix}$$

同理，可得接下来的两点为：

$$\boldsymbol{P}_2=\begin{bmatrix}1.900691\\0.311213\end{bmatrix}\quad 和\quad \boldsymbol{P}_3=\begin{bmatrix}1.900677\\0.311219\end{bmatrix}$$

$\boldsymbol{P}_3$ 的值的精度为小数点后 6 位。求解 $\boldsymbol{P}_2$ 和 $\boldsymbol{P}_3$ 的计算过程如表 3.7 所示。

表 3.7 用牛顿法求解例 3.32 过程中每个迭代的函数值、雅克比矩阵和微分值

$\boldsymbol{P}_k$	求解线法方程组 $\boldsymbol{J}(\boldsymbol{P}_k)\Delta\boldsymbol{P}=-\boldsymbol{F}(\boldsymbol{P}_k)$	$\boldsymbol{P}_k+\Delta\boldsymbol{P}$
$\begin{bmatrix}2.00\\0.25\end{bmatrix}$	$\begin{bmatrix}2.0 & -1.0\\4.0 & 2.0\end{bmatrix}\begin{bmatrix}-0.09375\\0.0625\end{bmatrix}=-\begin{bmatrix}0.25\\0.25\end{bmatrix}$	$\begin{bmatrix}1.90625\\0.3125\end{bmatrix}$
$\begin{bmatrix}1.90625\\0.3125\end{bmatrix}$	$\begin{bmatrix}1.8125 & -1.0\\3.8125 & 2.5\end{bmatrix}\begin{bmatrix}-0.005559\\-0.001287\end{bmatrix}=-\begin{bmatrix}0.008789\\0.024414\end{bmatrix}$	$\begin{bmatrix}1.900691\\0.311213\end{bmatrix}$
$\begin{bmatrix}1.900691\\0.311213\end{bmatrix}$	$\begin{bmatrix}1.801381 & -1.000000\\3.801381 & 2.489700\end{bmatrix}\begin{bmatrix}-0.000014\\0.000006\end{bmatrix}=-\begin{bmatrix}0.000031\\0.000038\end{bmatrix}$	$\begin{bmatrix}1.900677\\0.311219\end{bmatrix}$

实现牛顿法需要求解多个偏导数。可以利用数值逼近来近似这些偏导数，但必须注意确定适当的步长。在更高维数的情况，有必要利用本章前面讲述的线性方程组求解法求解 $\Delta\boldsymbol{P}$。

3.7.7 MATLAB

程序 3.6（非线性 Seidel 迭代）和程序 3.7（牛顿拉夫申法）需要分别将非线性方程组 $\boldsymbol{X}=\boldsymbol{G}(\boldsymbol{X})$、非线性方程组 $\boldsymbol{F}(\boldsymbol{X})=\boldsymbol{0}$ 和它的雅克比矩阵保存到 M 文件。例如，分别将例 3.32 中的非线性方程组和相关的雅克比矩阵保存到 F.m 和 JF.m 文件中。

```
function Z = F(X)
x = X(1);y = X(2);
Z = zeros(1,2);
Z(1) = x^2 - 2 * x - y + 0.5;
Z(2) = x^2 + 4y^2 - 4;
```

```
function W = JF(X)
x = X(1);y = X(2)
W = [2 * x - 2  - 1;2 * x 8 * y];
```

利用标准 MATLAB 命令计算这些函数：

```
> > A = feval('F',[2.00 0.25])
```

```
A=
   0.2500 0.2500
> >V=JF([2.00 0.25])
B=
   2 -1
   4 2
```

程序 3.6(非线性 Seidel 迭代) 求解非线性固定点方程组 $X=G(X)$,给定初始近似值 P_0,并生成序列 $\{P_k\}$ 收敛到解 P

```
function [P,iter] = seidel(G,P,delta,max1)
% Input   - G is the nonlinear system saved in the M-file G.m
%         - P is the initial guess at the solution
%         - delta is the error bound
%         - max1 is the number of iterations
% Output  - P is the seidel approximation to the solution
%         - iter is the number of iterations required
N=length(P);
for k=1:max1
   X=P;
   % X is the kth approximation to the solution
   for j=1:N
      A=feval('G',X);
      % Update the terms of X as they are calculated
      X(j)=A(j);
   end
   err=abs(norm(X-P));
   relerr=err/(norm(X)+eps);
   P=X;
   iter=k;
   if(err<delta)|(relerr<delta)
      break
   end
end
```

在下面的程序中,使用 MATLAB 命令 A \ B 求解线性方程组 $AX=B$(参见 $Q=P-(J\backslash Y')'$)。使用本章前面开发的程序代替这个 MATLAB 命令。选择适当的程序求解线性方程组依赖于雅克比矩阵的大小和特性。

程序 3.7(牛顿拉夫申法) 求解非线性方程组 $F(X)=0$,给定初始近似值 P_0,并生成序列 $\{P_k\}$ 收敛到解 P

```
function [P,iter,err]=newdim(F,JF,P,delta,epsilon,max1)
% Input   - F is the system saved as the M-file F.m
%         - JF is the Jacobian of F saved as the M-file JF.M
%         - P is the initial approximation to the solution
%         - delta is the tolerance for P
%         - epsilon is the tolerance for F(P)
%         - max1 is the maximum number of iterations
```

```
% Output  - P is the approximation to the solution
%         - iter is the number of iterations required
%         - err is the error estimate for P
Y = feval(F,P);

for k = 1:max1
   J = feval(JF,P);
   Q = P - (J \ Y')';
   Z = feval(F,Q);
   err = norm(Q - P);
   relerr = err/(norm(Q) + eps);
   P = Q;
   Y = Z;
   iter = k;
   if (err < delta)|(relerr < delta)|(abs(Y) < epsilon)
      break
   end
end
```

3.7.8 求解非线性方程组的迭代法的练习

1. 求解下列方程组的固定点：

   (a) $x = g_1(x, y) = x - y^2$

   $y = g_2(x, y) = -x + 6y$

   (b) $x = g_1(x, y) = (x^2 - y^2 - x - 3)/3$

   $y = g_2(x, y) = (-x + y - 1)/3$

   (c) $x = g_1(x, y = \sin(y)$

   $y = g_2(x, y) = -6x + y$

   (d) $x = g_1(x, y, z) = 9 - 3y - 2z$

   $y = g_2(x, y, z) = 2 - x + z$

   $z = g_3(x, y, z) = -9 + 3x + 4y - z$

2. 求解下列方程组的零点。计算每个方程组在零点处的雅克比矩阵：

   (a) $0 = f_1(x, y) = 2x + y - 6$

   $0 = f_2(x, y) = x + 2y$

   (b) $0 = f_1(x, y) = 3x^2 + 2y - 4$

   $0 = f_2(x, y) = 2x + 2y - 3$

   (c) $0 = f_1(x, y) = 2x - 4\cos(y)$

   $0 = f_2(x, y) = 4x\ \sin(y)$

   (d) $0 = f_1(x, y, z) = x^2 + y^2 - z$

   $0 = f_2(x, y, z) = x^2 + y^2 + z^2 - 1$

   $0 = f_3(x, y, z) = x + y$

3. 对下列方程组求解一个在 xy 平面的区间，如果(p_0, q_0)位于这个区间，则固定点迭代保证收敛(参见定理 3.17)：

$$x = g_1(x,y) = (x^2 - y^2 - x - 3)/3$$
$$y = g_2(x,y) = (x + y + 1)/3$$

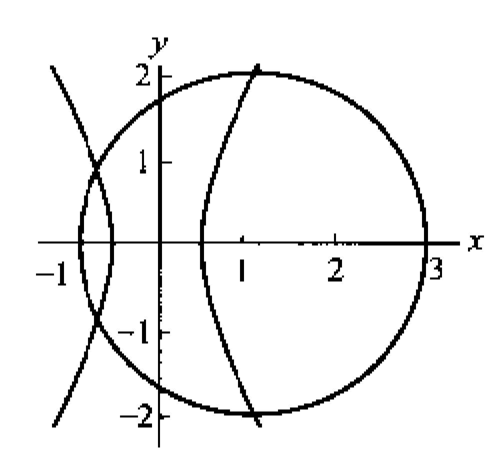


图 3.7　练习 5 中的双曲线和圆

4. 用固定点形式重写下列线性方程组。求 x, y, z 的边界，使得对于任意满足边界条件的初始值(p_0, q_0, r_0)，固定点迭代保证收敛：

$$\begin{aligned} 6x + y + z &= 1 \\ x + 4y + z &= 2 \\ x + y + 5z &= 0 \end{aligned}$$

5. 对下列给定的非线性方程组，分别采用(a) 固定点迭代与式(14)和(b) 利用方程式(18)的 Seidel 迭代。使用初始近似值$(p_0, q_0) = (1.1, 2.0)$，计算接下来的 3 个固定点近似值：

$$x = g_1(x,y) = \frac{8x - 4x^2 + y^2 + 1}{8} \qquad (\text{双曲线})$$
$$y = g_2(x,y) = \frac{2x - x^2 + 4y - y^2 + 3}{4} \qquad (\text{圆})$$

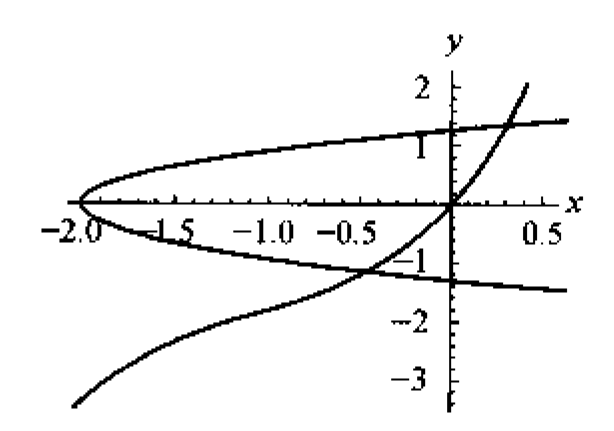


图 3.8　练习 6 中的三次曲线和双曲线

6. 对下列非线性方程组，分别采用(a) 固定点迭代与式(14)和(b) 利用式(18)的 Seidel 迭代。使用初始近似值$(p_0, q_0) = (-0.3, -1.3)$，求接下来的 3 个固定点的近似值：

$$x = g_1(x,y) = \frac{y - x^3 + 3x^2 + 3x}{7} \qquad (\text{三次曲线})$$
$$y = g_2(x,y) = \frac{y^2 + 2y - x - 2}{2} \qquad (\text{双曲线})$$

7. 设有非线性方程组：

$$0 = f_1(x,y) = x^2 - y - 0.2$$

$$0=f_2(x,y)=y^2-x-0.3$$

这些抛物线交于两点，如图 3.9 所示。

(a) 初始近似值 $(p_0,q_0)=(1.2,1.2)$，利用牛顿法计算 (p_1,q_1) 和 (p_2,q_2)。

(b) 初始近似值 $(p_0,q_0)=(-0.2,-0.2)$，利用牛顿法计算 (p_1,q_1) 和 (p_2,q_2)。

8. 设有下列非线性方程组，如图 3.10 所示：

$$0=f_1(x,y)=x^2+y^2-2$$
$$0=f_2(x,y)=xy-1$$

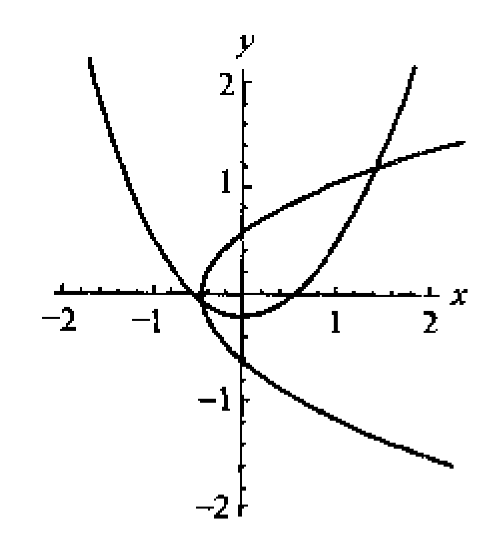


图 3.9　练习 7 中的抛物线

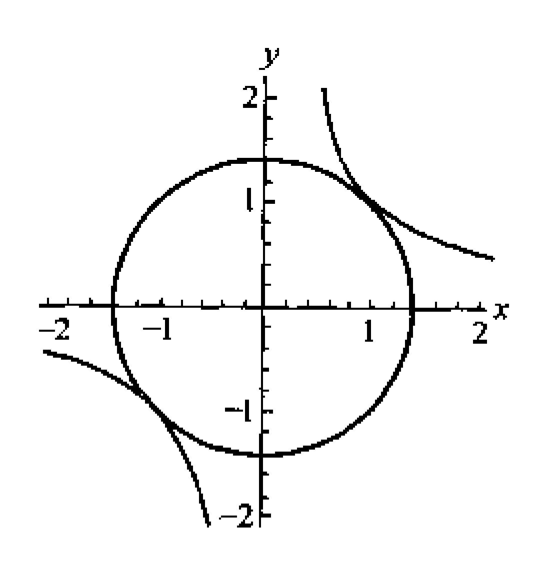


图 3.10　练习 8 中的圆和双曲线

(a) 验证解为 $(1,1)$ 和 $(-1,-1)$。

(b) 如果用牛顿法求解，存在什么样的困难?

9. 证明求解 3×3 线性方程组的雅克比迭代是固定点迭代(15)的一个特例。验证如果 3×3 线性方程组的系数矩阵具有严格对角优势，则满足条件(17)。

10. 证明求解两个方程的牛顿法可表示成固定点迭代的形式：

$$x=g_1(x,y),\quad y=g_2(x,y)$$

这里 $g_1(x,y)$ 和 $g_2(x,y)$ 表示为：

$$g_1(x,y)=x-\frac{f_1(x,y)\frac{\partial}{\partial y}f_2(x,y)-f_2(x,y)\frac{\partial}{\partial y}f_1(x,y)}{\det(\boldsymbol{J}(x,y))}$$

$$g_2(x,y)=y-\frac{f_2(x,y)\frac{\partial}{\partial x}f_1(x,y)-f_1(x,y)\frac{\partial}{\partial x}f_2(x,y)}{\det(\boldsymbol{J}(x,y))}$$

11. 用固定点迭代求解非线性方程组(12)。使用下面的步骤证明式(16)中的条件是保证 $\{(p_k,q_k)\}$ 收敛到 (p,q) 的充分条件。设有常量 K，$0<K<1$，因此对位于矩形区域 $R=\{(x,y):a<x<b,c<y<d\}$ 中的所有 (x,y)，有：

$$\left|\frac{\partial}{\partial x}g_1(x,y)\right|+\left|\frac{\partial}{\partial y}g_1(x,y)\right|<K$$

且：

$$\left|\frac{\partial}{\partial x}g_2(x,y)\right|+\left|\frac{\partial}{\partial y}g_2(x,y)\right|<K$$

假设 $a<p_0<b$ 且 $c<q_0<d$。定义：

$$e_k=p-p_k,\quad E_k=q-q_k\quad 和\quad r_k=\max\{|e_k|,|E_k|\}$$

对有两个变量的函数利用如下形式的均值定理:

$$e_{k+1}=\frac{\partial}{\partial x}g_1(a_k^*,q_k)e_k+\frac{\partial}{\partial y}g_1(p,c_k^*)E_k$$

$$E_{k+1}=\frac{\partial}{\partial x}g_2(b_k^*,q_k)e_k+\frac{\partial}{\partial y}g_2(p,d_k^*)E_k$$

这里 a_k^* 和 b_k^* 位于$[a,b]$,而且 c_k^* 和 d_k^* 位于$[c,d]$。证明下列命题:

(a) $|e_1|\leqslant Kr_0$且$|E_1|\leqslant Kr_0$

(b) $|e_2|\leqslant Kr_1\leqslant K^2r_0$且$|E_2|\leqslant Kr_1\leqslant K^2r_0$

(c) $|e_k|\leqslant Kr_{k-1}\leqslant K^kr_0$且$|E_k|\leqslant Kr_{k-1}\leqslant K^kr_0$

(d) $\lim_{n\to\infty}p_k=p$ 且 $\lim_{n\to\infty}q_k=q$

12. 正如前面指出的,方程组(20)的雅克比矩阵是导数的二维模拟近似。将方程组(20)表示成向量函数 $\boldsymbol{V}=\boldsymbol{F}(\boldsymbol{X})$,而且将 $\boldsymbol{J}(\boldsymbol{F})$作为这个方程组的雅克比矩阵。给定两个非线性方程组 $\boldsymbol{V}=\boldsymbol{F}(\boldsymbol{X})$和 $\boldsymbol{V}=\boldsymbol{G}(\boldsymbol{X})$,并且给定实数 c,证明:

(a) $\boldsymbol{J}(c\boldsymbol{F}(\boldsymbol{X}))=c\boldsymbol{J}(\boldsymbol{F}(\boldsymbol{X}))$

(b) $\boldsymbol{J}(\boldsymbol{F}(\boldsymbol{X})+\boldsymbol{G}(\boldsymbol{X}))=\boldsymbol{J}(\boldsymbol{F}(\boldsymbol{X}))+\boldsymbol{J}(\boldsymbol{G}(\boldsymbol{X}))$

3.7.9 算法和程序

1. 使用程序 3.6 求解练习 5 和练习 6 中方程组的固定点近似值,结果精确到小数点后 10 位。
2. 使用程序 3.7 求解练习 7 和练习 8 中方程组的零点近似值,结果精确到小数点后 10 位。
3. 构造一个程序,利用固定点迭代求解方程组的固定点。使用此程序求解练习 5 和练习 6 中方程组的固定点近似值,结果精确到小数点后 8 位。
4. 使用程序 3.7 求解下列方程组的零点近似值,结果精确到小数点后 10 位:

(a)
$$0=x^2-x+y^2+z^2-5$$
$$0=x^2+y^2-y+z^2-4$$
$$0=x^2+y^2+z^2+z-6$$

(b)
$$0=x^2-x+2y^2+yz-10$$
$$0=5x-6y+z$$
$$0=z-x^2-y^2$$

(c)
$$0=(x+1)^2+(y+1)^2-z$$
$$0=(x-1)^2+y^2-z$$
$$0=4x^2+2y^2+z^2-16$$

(d)
$$0=9x^2+36y^2+4z^2-36$$
$$0=x^2-2y^2-20z$$
$$0=16x-x^3-2y^2-16z^2$$

5. 为了求解下列非线性方程组:

$$0=7x^3-10x-y-1$$
$$0=8y^3-11y+x-1$$

使用 MATLAB 在同一坐标画出两个曲线。根据画出的图验证两个曲线有 9 点相交,并估计相交点坐标。根据这些估计值,使用程序 3.7 求解这些点的近似值,精确到小数点后 9 位。

6. 问题 5 中的方程组可表示为固定点的形式:

$$x = \frac{7x^3 - y - 1}{10}$$

$$y = \frac{8y^3 + x - 1}{11}$$

通过使用计算机进行练习可以发现,无论使用什么初始近似值,利用固定点迭代(利用这个特殊的固定点形式)只能找到 9 个相交点中的一个。请问是否存在其他的固定点形式,可求解方程组中的其他解?

第4章 插值与多项式逼近

用于评价库函数，如 $\sin(x)$，$\cos(x)$，e^x 的计算机软件所使用的计算过程用到了多项式逼近。目前最先进的方法使用有理函数(即多项式的商)，但多项式逼近理论适合作为数值分析的入门课程，因此我们在这一章中对它进行回顾。设在区间$[-1,1]$内对函数 $f(x)=e^x$ 进行二次多项式逼近，图 4.1(a)为其泰勒多项式结果，图 4.1(b)为其切比雪夫多项式结果。泰勒多项式的最大误差为 0.218282，而切比雪夫多项式的最大误差为 0.056468。本章将推导出考察这些问题所需的基本定理。

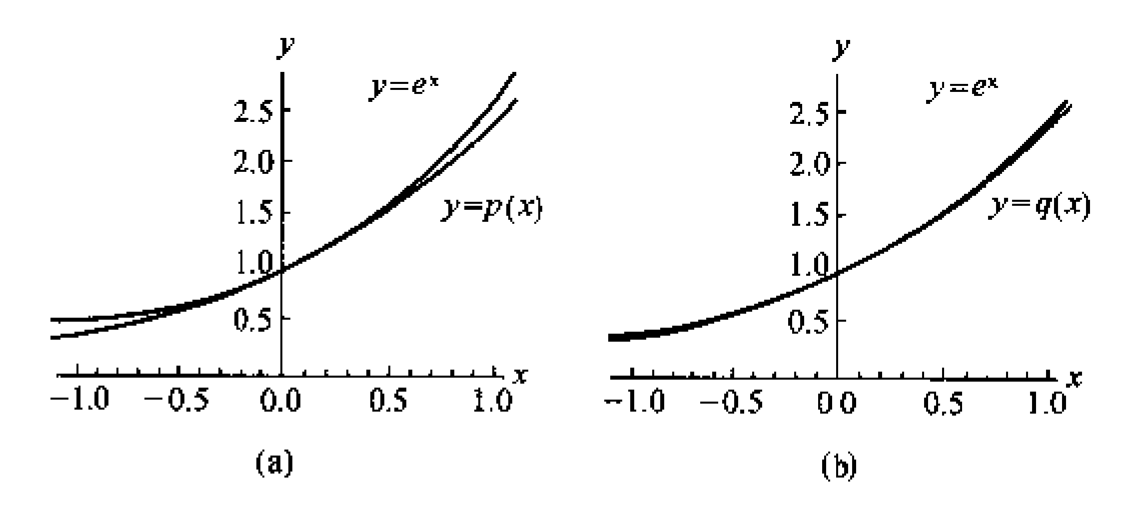


图 4.1 (a)区间$[-1,1]$内 $p(x)=1.000000+1.000000x+0.500000x^2$ 的泰勒多项式 $f(x)=e^x$ 逼近
(b)区间$[-1,1]$内 $f(x)=e^x$ 的切比雪夫多项式 $q(x)=1.000000+1.129772x+0.532042x^2$ 逼近

在组合多项式构造中有一个相关的问题：给定平面上的 $n+1$ 个点(其中任意两点都不在一条垂直线上)，组合多项式是过这些点的、惟一且次数小于等于 n 的多项式。在已知数据具有高精度的情况下，组合多项式有时用来寻找通过给定数据点的多项式。构造组合多项式的方法有许多种：线性方程求解、使用拉格朗日(Lagrange)系数多项式、构造分段差分表和牛顿多项式系数，上述三者都是数值分析中的重要技术。例如，过点(1,2)，(2,1)，(3,5)，(4,6)，(5,1)的组合多项式为：

$$P(x)=\frac{5x^4-82x^3+427x^2-806x+504}{24}$$

图 4.2 中显示了点及多项式曲线。

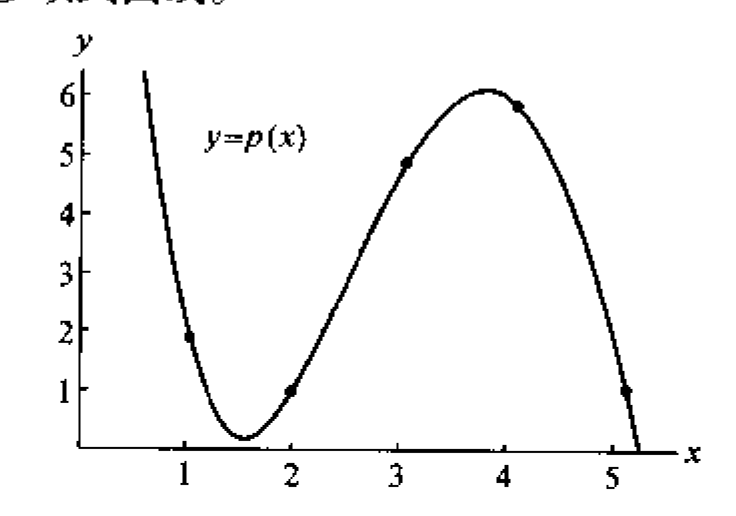


图 4.2 过点(1,2)，(2,1)，(3,5)，(4,6)，(5,1)的组合多项式图

4.1 泰勒级数和函数计算

极限过程是微积分的基础,例如,导数:

$$f'(x)=\lim_{h\to 0}\frac{f(x+h)-f(x)}{h}$$

是差商在分子分母均趋于 0 时的极限。泰勒级数是另一类型的极限过程,即无穷多项相加,并求部分和的极限。其重要应用之一是表示基本函数:$\sin(x)$,$\cos(x)$,e^x,$\ln(x)$等。表 4.1 是一些常用的泰勒级数展开。部分和可求到满足指定的精度要求为止,级数方法通常用于工程和物理领域中。

现在来看怎样由有限和得到无限和的较好逼近。作为示例,我们使用表 4.1 中的指数级数来计算自然对数和指数函数的基,$e=e^1$。选择 $x=1$,并计算级数:

$$e^1=1+\frac{1}{1!}+\frac{1^2}{2!}+\frac{1^3}{3!}+\frac{1^4}{4!}+\cdots+\frac{1^k}{k!}+\cdots$$

表 4.1 一些常函数的泰勒级数展开

$\sin(x)=x-\frac{x^3}{3!}+\frac{x^5}{5!}-\frac{x^7}{7!}+\cdots$	对所有 x
$\cos(x)=1-\frac{x^2}{2!}+\frac{x^4}{4!}-\frac{x^6}{6!}+\cdots$	对所有 x
$e^x=1+x+\frac{x^2}{2!}+\frac{x^3}{3!}+\frac{x^4}{4!}+\cdots$	对所有 x
$\ln(1+x)=x-\frac{x^2}{2!}+\frac{x^3}{3!}-\frac{x^4}{4!}+\cdots$	$-1\leqslant x\leqslant 1$
$\arctan(x)=x-\frac{x^3}{3!}+\frac{x^5}{5!}-\frac{x^7}{7!}+\cdots$	$-1\leqslant x\leqslant 1$
$(1+x)^p=1+px+\frac{p(p-1)}{2!}x^2+\frac{p(p-1)(p-2)}{3!}x^3+\cdots$	$\lvert x\rvert\leqslant 1$

1.1 节中对无穷级数的和的定义要求部分和 S_k 收敛于一极限。表 4.2 中列出了这些和的值。

表 4.2 计算 e 的部分和 S_n

n	$S_n=1+\frac{1}{1!}+\frac{1}{2!}+\cdots+\frac{1}{n!}$
0	1.0
1	2.0
2	2.5
3	2.666666666666...
4	2.708333333333...
5	2.716666666666...
6	2.718055555555...
7	2.718253968254...
8	2.718278769841...
9	2.718281525573...
10	2.718281801146...
11	2.718281826199...
12	2.718281828286...
13	2.718281828447...
14	2.718281828458...
15	2.718281828459...

一种自然的考虑方法,是将函数的幂级数表示看作一种次数递增多项式的极限情况:若有足够的项相加,则可得到精确的逼近。这一点需要精确化,选择什么次数?怎样计算多项式中的系数?定理4.1将回答这些问题。

定理4.1(泰勒多项式逼近) 设$f\in C^{N+1}[a,b]$,而$x_0\in[a,b]$为固定值。若$x\in[a,b]$,则有:

$$f(x)=P_N(x)+E_N(x) \tag{1}$$

其中$P_N(x)$为一多项式,可用来近似$f(x)$:

$$f(x)\approx P_N(x)=\sum_{k=0}^{N}\frac{f^{(k)}(x_0)}{k!}(x-x_0)^k \tag{2}$$

误差项$E_N(x)$为:

$$E_N(x)=\frac{f(N+1)(c)}{(N+1)!}(x-x_0)^{N+1} \tag{3}$$

$c=c(x)$为x和x_0之间的某点。

证明留作练习。

(2)式说明如何计算泰勒多项式系数,虽然误差项(3)中有一个类似的项,注意$f^{(N+1)}(c)$在不确定点c处求值,c依赖于值x。因此我们不对$E_N(x)$求值,只用它来确定一个界,作为逼近的精确度。

例4.1 说明为什么要得到表4.2中13位数字的近似$e=2.718281828459$,只需15项。

在点$x_0=0$,将$f(x)=e^x$展开为15阶泰勒多项式,其中指数项为$(x-0)^k=x^k$,而导数项为$f'(x)=f''(x)=\cdots=f^{(16)}=e^x$,前15项导数用于计算系数$a_k=e^0/k!$,可写为:

$$P_{15}(x)=1+x+\frac{x^2}{2!}+\frac{x^3}{3!}+\cdots+\frac{x^{15}}{15!} \tag{4}$$

在式(4)中令$x=1$,得到部分和$S_{15}=P_{15}(1)$。剩余项用于计算逼近的精度:

$$E_{15}(x)=\frac{f^{(16)}(c)\ x^{16}}{16!} \tag{5}$$

由于选择了$x_0=0$和$x=1$,因此值c位于它们之间(即$0<c<1$),故$e^c<e^1$。注意表4.2中的部分和上限为3,两不等式合并得$e^c<3$,代入下面的计算,有:

$$|E_{15}(1)|=\frac{|f^{(16)}(c)|}{16!}\leqslant\frac{e^c}{16!}<\frac{3}{16!}<1.433844\times10^{-13}$$

从而,在近似值$e\approx2.718281828459$中的每一位都是有效数字,因为实际误差(不管是多少)在小数点后第13位必然小于2。

下面将讨论有关逼近的一些特点,而不给出定理4.1的严格证明(读者可在任何一本微积分标准教材中找到更多详细的讨论)。再次以函数$f(x)=e^x$和值$x_0=0$为例,由基本微积分可知,在点(x,e^x)处,曲线$y=e^x$处的斜率是$f'(x)=e^x$,故点(0,1)处曲线的切线为$f'(0)=1$,点(0,1)处曲线的切线为$y=1+x$;在定理4.1中用$N=1$可得出相同公式,即$P_1(x)=f(0)+f'(0)x/1!=1+x$,因此$P_1(x)$为该曲线的切线,见图4.3。

可以看出逼近$e^x\approx1+x$在中心$x_0=0$附近较好,而随着x远离0点,两条曲线之间的距

离增加。注意两条曲线在(0,1)处的斜率相等，在微积分中可知一条曲线的二阶导数指示它是上凹或下凹。若曲线 $y=f(x)$ 和 $y=g(x)$ 满足 $f(x_0)=g(x_0)$，$f'(x_0)=g'(x_0)$ 和 $f''(x_0)=g''(x_0)$，则它们在 x_0 处有相同曲率①。对于逼近函数 $f(x)$ 的多项式，这是一个良好的属性，推论 4.1 说明，对 N≥2，泰勒多项式具有这一属性。

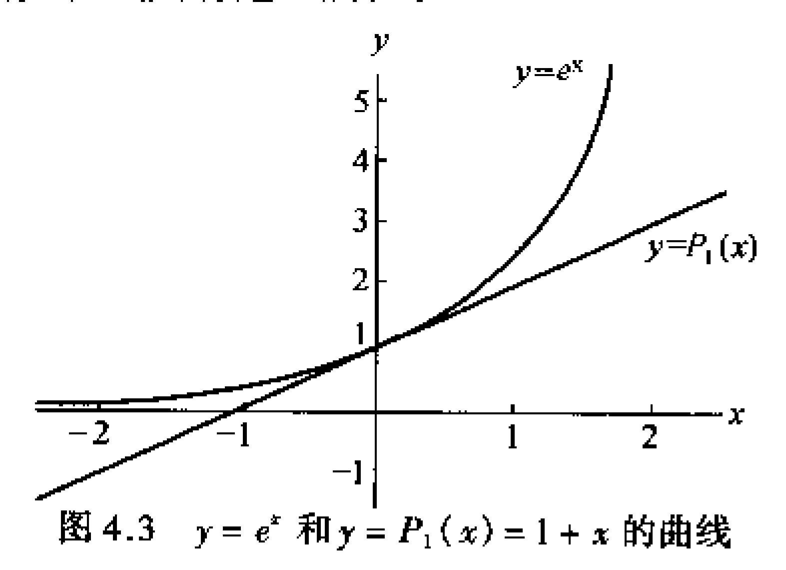


图 4.3　$y=e^x$ 和 $y=P_1(x)=1+x$ 的曲线

推论 4.1　若 P_N 为定理 4.1 给出的 N 次泰勒多项式，则：

$$P_N^{(k)}(x_0)=f^{(k)}(x_0),\ k=0,1,\cdots,N \tag{6}$$

证明：令式(2)和式(3)中 $x=x_0$，结果为 $P_N(x_0)=f(x_0)$，则当 $k=0$ 时(6)式成立。对(2)式右端求导，得：

$$P'_N(x)=\sum_{k=1}^{N}\frac{f^{(k)}(x_0)}{(k-1)!}(x-x_0)^{k-1}=\sum_{k=0}^{N-1}\frac{f^{(k+1)}(x_0)}{k!}(x-x_0)^{k} \tag{7}$$

令(7)中 $x=x_0$，则得 $P'_N(x_0)=f'(x_0)$，故当 $k=1$ 时(6)式成立。对(7)式连续求导可证明(6)式对其他情况也成立，详细证明过程留作练习。

由推论 4.1 可知，$y=P_2(x)$ 具有属性 $f(x_0)=P_2(x_0)$，$f'(x_0)=P'_2(x_0)$ 和 $f''(x_0)=P''_2(x_0)$，故它们在 x_0 处有相同曲率。例如，图 4.4 为 $f(x)=e^x$ 和 $P_2(x)=1+x+x^2/2$，可以看出在(0,1)点处它们有相同的曲线形态。

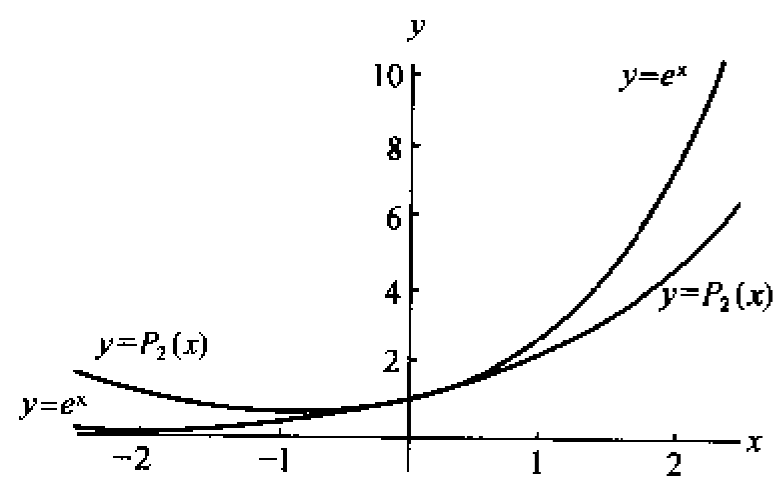


图 4.4　$y=e^x$ 和 $y=P_2(x)=1+x+x^2/2$ 曲线

在逼近理论中，我们总是试图寻找解析函数② $f(x)$ 在区间 $[a,b]$ 内的精确多项式逼近，这是开发计算机软件时使用的技术之一。泰勒多项式的精度随 N 的增长而提高；而通常，任何给定

① 曲线 $y=f(x)$ 在 (x_0,y_0) 处的曲率 K 定义为 $K=|f''(x_0)|/(1+[f'(x_0)]^2)^{3/2}$。

② 函数 $f(x)$ 在 x_0 处是解析的，若它有连续的各阶导数，则在 x_0 附近的一个区间中可表示为泰勒级数。

多项式的精度都将随 x 远离中心点 x_0 而降低。因此,我们必须选择足够大的 N,并限制最大值 $|x-x_0|$,使得误差不会超过给定限度。若选择区间宽度为 $2R$,而 x_0 位于区间中心(即 $|x-x_0|<R$),则误差绝对值满足关系:

$$|误差| = |E_N(x)| \leqslant \frac{MR^{N+1}}{(N+1)!} \tag{8}$$

其中 $M \leqslant \max\{|f^{(N+1)}(z)|: x_0-R \leqslant z \leqslant x_0+R\}$。若所有导数一致有界,则式(8)中的误差界与 $R^{N+1}/(N+1)$成正比,并且在 N 增加而 R 固定时或 N 固定而 R 趋于0时递减。表4.3显示了这两个参数的选择对区间 $|x| \leqslant R$ 内逼近 $e^x \approx P_N(x)$的精度影响,当 N 最大而 R 最小时误差最小。图4.5给出了 P_2,P_3,P_4 的曲线。

表4.3　在区间 $|x| \leqslant R$ 内逼近 $e^x \approx P_N(x)$的误差限 $|误差| < e^R R^{N+1}/(N+1)$的值

	$R=2.0, \|x\| \leqslant 2.0$	$R=1.5, \|x\| \leqslant 1.5$	$R=1.0, \|x\| \leqslant 1.0$	$R=0.5, \|x\| \leqslant 0.5$
$e^x \approx P_5(x)$	0.65680499	0.07090172	0.00377539	0.00003578
$e^x \approx P_6(x)$	0.18765857	0.01519323	0.00053934	0.00000256
$e^x \approx P_7(x)$	0.04691464	0.00284873	0.00006742	0.00000016
$e^x \approx P_8(x)$	0.01042548	0.00047479	0.00000749	0.00000001

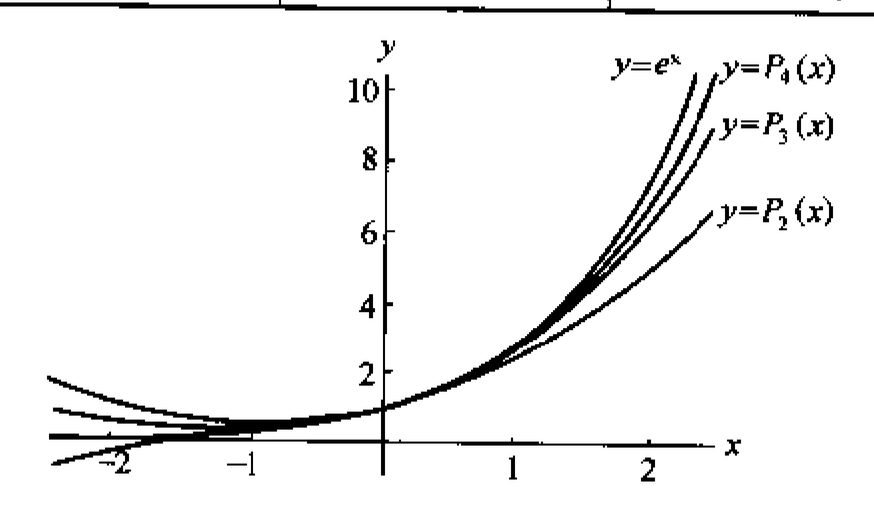


图4.5　$y=e^x$, $y=P_2(x)$, $y=P_3(x)$和 $y=P_4(x)$的曲线

例4.2　求逼近多项式 $e^x \approx P_8(x)$在区间 $|x| \leqslant 1.0$ 和 $|x| \leqslant 0.5$ 内的误差界。

若 $|x| \leqslant 1.0$,则令 $R=1.0$,由(8)式中的 $|f^{(9)}(c)| = |e^c| \leqslant e^{1.0} = M$ 有:

$$|\text{error}| = |E_8(x)| \leqslant \frac{e^{1.0}(1.0)^9}{9!} \approx 0.00000749$$

若 $|x| \leqslant 0.5$,则令 $R=0.5$,由(8)式中的 $|f^{(9)}(c)| = |e^c| \leqslant e^{0.5} = M$ 有:

$$|\text{error}| = |E_8(x)| \leqslant \frac{e^{0.5}(0.5)^9}{9!} \approx 0.00000001$$

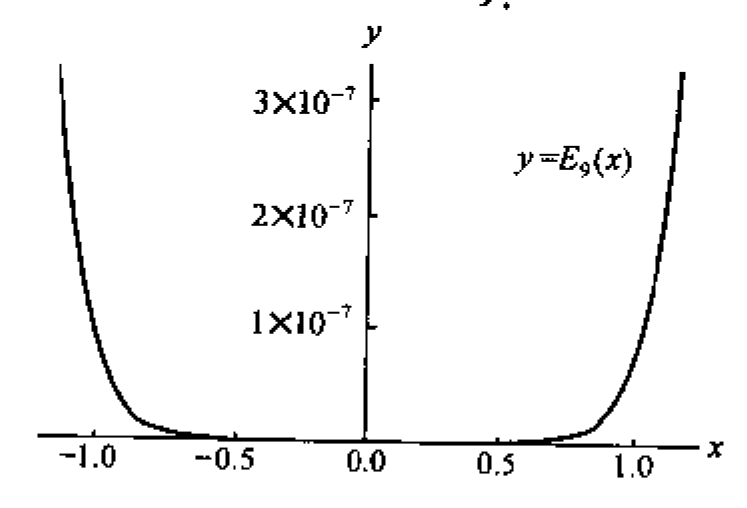


图4.6　误差 $y=E_9(x)=e^x-P_9(x)$曲线

例 4.3　若 $f(x)=e^x$，证明 $N=9$ 是满足区间 $[-1,1]$ 内 $|误差|=|E_N(x)|\leqslant 0.0000005$ 的最小整数。因此 $P_9(x)$ 可用来计算 e^x 的近似值，精确到小数点后第 6 位。需要找到满足的最小整数。

$$|\text{error}|=|E_N(x)|\leqslant \frac{e^c(1)^{N+1}}{(N+1)!}<0.0000005$$

由例 4.2 知，$N=8$ 太小，故试用 $N=9$，并发现 $|E_N(x)|\leqslant e^1(1)^{9+1}/(9+1)!\ \leqslant 0.000000749$，该值略大于需要值，故我们很可能选择 $N=10$。但在确定误差限时我们使用的是 $e^c\leqslant e^1$ 作为粗略估计，因此 0.000000749 只比实际误差稍大了一点。图 4.6 显示了 $E_9(x)=e^x-P_9(x)$ 的曲线，注意最大垂直距离约为 3×10^{-7}，在右端点 $(1,E_9(1))$ 处。实际上，该区间内的最大误差为 $E_9(1)=2.718281828-2.718281526\approx 3.024\times10^{-7}$，因此，$N=9$ 就够了。

4.1.1　多项式计算方法

计算多项式有多种数学上等价的方法。例如，考虑函数：

$$f(x)=(x-1)^8 \tag{9}$$

$f(x)$ 的计算需要用到指数函数，或用二项式公式将 $f(x)$ 展开为 x 的幂：

$$\begin{aligned} f(x) &= \sum_{k=0}^{8}\binom{8}{k}x^{8-k}(-1)^k \\ &= x^8-8x^7+28x^6-56x^5+70x^4-56x^3+28x^2-8x+1 \end{aligned} \tag{10}$$

霍纳(Horner)方法(见 1.1 节)，也称为嵌套乘法，可以用来计算式(10)中的多项式，式(10)用霍纳方法改写为：

$$f(x)=(((((((x-8)x+28)x-56)x+70)x-56)x+28)x-8)x+1 \tag{11}$$

这样计算 $f(x)$ 需要 7 个乘法和 8 个加(减)法，从而消除了指数函数的计算。

我们以一个与表 4.1 中的泰勒级数和定理 4.1 中的泰勒多项式相关的定理来结束本节。

定理 4.2(泰勒级数)　设 $f(x)$ 是解析的，并在包含 x_0 的一个区间 (a,b) 有连续的各阶导数 $N=1,2,\cdots$。设泰勒多项式(2)趋近于一个极限：

$$S(x)=\lim_{N\to\infty}P_N(x)=\lim_{N\to\infty}\sum_{k=0}^{N}\frac{f^{(k)}(x_0)}{k!}(x-x_0)^k \tag{12}$$

则 $f(x)$ 有泰勒级数展开：

$$f(x)=\sum_{k=0}^{\infty}\frac{f^{(k)}(x_0)}{k!}(x-x_0)^k \tag{13}$$

证明：直接由节 1.1 中的级数收敛定义得到。极限条件通常描述为当 N 趋于无穷大时，误差项趋于 0，因此式(13)成立的一个充要条件是：

$$\lim_{N\to\infty}E_N(x)=\lim_{N\to\infty}\frac{f^{(N+1)}(c)\ (x-x_0)^{N+1}}{(N+1)!}=0 \tag{14}$$

其中 c 依赖于 N 和 x。

4.1.2　习题

1.　设 $f(x)=\sin(x)$，应用定理 4.1，

(a) 对 $x_0=0$,计算 $P_5(x)$,$P_7(x)$和 $P_9(x)$。

(b) 证明:若$|x|\leqslant 1$,则逼近多项式:

$$\sin(x)\approx x-\frac{x^3}{3!}+\frac{x^5}{5!}+\frac{x^7}{7!}+\frac{x^9}{9!}$$

有误差限$|E_9(x)|<1/10!\ \leqslant 2.75574\times 10^{-7}$。

(c) 用 $x_0=\pi/4$,计算 $P_5(x)$,其中包含$(x-\pi/4)$的幂函数。

2. 设$f(x)=\cos(x)$,使用定理 4.1,

(a) 对 $x_0=0$,计算 $P_4(x)$,$P_6(x)$和 $P_8(x)$。

(b) 证明:若$|x|\leqslant 1$,则逼近多项式:

$$\cos(x)\approx 1-\frac{x^2}{2!}+\frac{x^4}{4!}-\frac{x^6}{6!}+\frac{x^8}{8!}$$

有误差限$|E_8(x)|<1/9!\ \leqslant 2.75574\times 10^{-6}$。

(c) 用 $x_0=\pi/4$,计算 $P_4(x)$,其中包含$(x-\pi/4)$)的幂函数。

3. 函数$f(x)=x^{1/2}$在点 $x_0=0$ 和$f(x)=x^{1/2}$附近是否存在泰勒级数展开? 试证明你的结论。

4. (a) 求函数$f(x)=1/(1+x)$在 $x_0=0$ 附近 $N=5$ 的泰勒多项式。

(b) 求(a)中的逼近多项式的误差项 $E_5(x)$。

5. 求函数$f(x)=e^{-x^2/2}$在 $x_0=0$ 附近 $N=3$ 的泰勒多项式。

6. 求函数$f(x)=x^3-2x^2+2x$ 在 $x_0=1$ 附近 $N=3$ 的泰勒多项式 $P_3(x)$,并证明 $f(x)=P_3(x)$。

7. (a) 求函数$f(x)=x^{1/2}$在 $x_0=4$ 附近 $N=5$ 的泰勒多项式。

(b) 求函数$f(x)=x^{1/2}$在 $x_0=9$ 附近 $N=5$ 的泰勒多项式。

(c) 判断(a)和(b)中哪个多项式更好地逼近$(6.5)^{1/2}$。

8. 对$f(x)=(2+x)^{1/2}$,使用定理 4.1,

(a) 求 $x_0=2$ 附近的泰勒多项式 $P_3(x)$。

(b) 用 $P_3(x)$计算 $3^{1/2}$的近似值。

(c) 求区间 $1\leqslant c\leqslant 3$ 内$|f^{(4)}(c)|$的最大值,并计算$|E_3(x)|$的界。

9. 求在 $x_0=0$ 附近需要展开的泰勒多项式 $P_N(x)$的次数,使得对 $e^{0.1}$的逼近误差小于 10^{-6}。

10. 求在 $x_0=\pi$ 附近需要展开的泰勒多项式 $P_N(x)$的次数,使得对 $\cos(33\pi/32)$的逼近误差小于 10^{-6}。

11. (a) 求 $F(x)=\int_{-1}^{x}\cos(t^2)\,dt$ 在 $x_0=0$ 附近 $N=4$ 的泰勒多项式。

(b) 用泰勒多项式求 $F(0.1)$的近似值。

(c) 求(b)中近似计算的误差界。

12. (a) 对$|x|<1$ 区间内的几何级数:

$$\frac{1}{1+x^2}=1-x^2+x^4-x^6+x^8-\cdots,\text{其中}|x|<1$$

两端逐项积分,得:

$$\arctan(x)=x-\frac{x^3}{3}+\frac{x^5}{5}-\frac{x^7}{7}+\cdots,\text{其中}|x|<1$$

(b) 利用 $\pi/6 = \arctan(3^{-1/2})$和(a)中的级数，证明：

$$\pi = 3^{1/2} \times 2\left(1 - \frac{3^{-1}}{3} + \frac{3^{-2}}{5} - \frac{3^{-3}}{7} + \frac{3^{-4}}{9} - \cdots\right)$$

(c) 利用(b)中的级数，计算精确到 8 位数字的 π 值：

$$\pi \approx 3.141592653589793284\ldots$$

13. 利用 $f(x) = \ln(1+x)$和 $x_0 = 0$，并使用定理 4.1，

(a) 证明 $f^{(k)}(x) = (-1)^{k-1}((k-1)!)/(1+x)^k$

(b) 证明 N 次泰勒多项式为：

$$P_N(x) = x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \cdots + \frac{(-1)^{N-1}x^N}{N}$$

(c) 证明 $P_N(x)$的误差项为：

$$E_N(x) = \frac{(-1)^N x^{N+1}}{(N+1)(1+c)^{N+1}}$$

(d) 计算 $P_3(0.5)$，$P_6(0.5)$，$P_9(0.5)$，并与 $\ln(1.5)$进行比较。

(e) 证明若 $0.0 \leqslant x \leqslant 0.5$，则逼近多项式：

$$\ln(x) \approx x - \frac{x^2}{2} + \frac{x^3}{3} - \cdots + \frac{x^7}{7} - \frac{x^8}{8} + \frac{x^9}{9}$$

具有误差界 $|E_9| \leqslant 0.00009765$。

14. 二项式级数。设 $f(x) = (1+x)^p$，且 $x_0 = 0$。

(a) 证明 $f^{(k)}(x) = p(p-1)\cdots(p-k+1)(1+x)^{p-k}$。

(b) 证明其 N 次泰勒多项式为：

$$P_N(x) = 1 + px + \frac{p(p-1)x^2}{2!} + \cdots + \frac{p(p-1)\cdots(p-N+1)x^N}{N!}$$

(c) 证明：

$$E_N(x) = p(p-1)\cdots(p-N)x^{N+1}/((1+c)^{N+1-p}(N+1)!)$$

(d) 令 $p = 1/2$，计算 $P_2(0.5)$，$P_4(0.5)$和 $P_6(0.5)$，并与$(1.5)^{1/2}$进行比较。

(e) 证明若 $0.0 \leqslant x \leqslant 0.5$，则逼近多项式：

$$(1+x)^{1/2} \approx 1 + \frac{x}{2} - \frac{x^2}{8} + \frac{x^3}{16} - \frac{5x^4}{128} + \frac{7x^5}{256}$$

有误差界 $|E_5| \leqslant (0.5)^6(21/1024) = 0.0003204\cdots$

(f) 证明若 $p = N$ 为一正整数，则：

$$P_N(x) = 1 + Nx + \frac{N(N-1)x^2}{2!} + \cdots + Nx^{N-1} + x^N$$

注意这是著名的二项式展开。

15. 求解 c，使得对任意 $|E_4| \leqslant 10^{-6}$，有 $|x - x_0| < c$。

(a) 设 $f(x) = \cos(x)$，且 $x_0 = 0$。

(b) 设 $f(x) = \sin(x)$，且 $x_0 = \pi/2$。

(c) 设 $f(x) = e^x$，且 $x_0 = 0$。

16. (a) 设 $y = f(x)$为一偶函数(即，对于 f 定义域内的所有 x，$f(-x) = f(x)$)，$P_N(x)$具有什么性质？

(b) 设 $y=f(x)$为一奇函数(即,对于 f 定义域内的所有 x,$f(-x)=-f(x)$),$P_N(x)$ 具有什么性质?

17. 设 $y=f(x)$为一 N 次多项式,若 $f(x_0)>0$,且 $f'(x_0),\cdots,f^{(N)}(x_0)\geqslant 0$,证明:$f$ 的所有实根小于 x_0。提示:将 f 在 x_0 附近展开为泰勒多项式。
18. 设 $f(x)=e^x$,利用定理 4.1 计算 $x_0=0$ 附近的 $P_N(x)$,$N=1,2,3\cdots$。证明:$P_N(x)$的每个实根有小于等于 1 的重数。注意:若 p 为多项式 $P(x)$的 M 重根,则它是 $P'(x)$ 的 $M-1$ 重根。
19. 通过 $P_N^{(k)}(x)$的表达式和下式:

$$P_N^{(k)}(x_0)=f^{(k)}(x_0),\ \ k=2,3,\cdots,N$$

完成推论 4.1 的证明。

习题 20 和 21 完成对泰勒定理的证明。

20. 设 $g(t)$及其导数 $g^{(k)}(t)$,$k=1,2,\cdots,N+1$ 在区间(a,b)内连续,x_0 为区间内一点。若存在两个不同的点 x 和x_0,满足 $g(x)=0$,且 $g(x_0)=g'(x_0)=\cdots g^{(N)}(x_0)$,证明:存在一个值 c 在 x 和x_0 之间,且 $g^{(N+1)}(c)=0$。

说明:注意 $g(t)$为 t 的函数,值 x 和x_0 应看作与变量 t 相关的常数。

提示:利用罗尔定理(第 1.1 节,定理 1.5),在以 x_0 和 x 为端点的闭区间内找到点 c_1,满足 $g'(c_1)=0$。再对 $g'(t)$和以 x_0、c_1 为端点的区间应用罗尔定理,找到满足式 c_2 的数 $g''(c_2)$。依此类推,直到找到 c_{N+1},满足 $g^{(N+1)}(c_{N+1})=0$。
21. 利用练习 20 的结果和函数:

$$g(t)=f(t)-P_N(t)-E_N(x)\frac{(t-x_0)^{N+1}}{(x-x_0)^{N+1}}$$

其中 $P_N(x)$为 N 次泰勒多项式,证明误差项 $E_N(x)=f(x)-P_N(x)$形如:

$$E_N(x)=f^{(N+1)}(c)\ \frac{(x-x_0)^{N+1}}{(N+1)!}$$

提示:找出 $g^{(N+1)}(t)$,并求其在 $t=c$ 处的值。

4.1.3 算法与程序

MATLAB 的矩阵特性使我们能够快速地计算一个函数在多个点处的值,例如,若 X = [-1 0 1],则 sin(X)将产生[sin(-1) sin(0) sin(1)]。类似地,若 X = -1:0.1:1,则 Y = X 将得到一个与 X 同样维数的矩阵 Y,其值为正弦函数的值。通过定义矩阵 D = [X'Y'],这两个行矩阵可以输出为表(注意:矩阵 X 和 Y 必须有相同的长度)。

1. (a) 用 plot 命令,在同一幅图中绘制区间 $-1\leqslant x\leqslant 1$ 内的 $\sin(x)$,以及习题 1 中计算出的$P_5(x)$,$P_7(x)$和 $P_9(x)$。

   (b) 建立一个表,其列为在区间$[-1,1]$内以等距的 10 个点 x 上的 $\sin(x)$,$P_5(x)$,$P_7(x)$,$P_9(x)$。
2. (a) 用 plot 命令在同一幅图中绘制出区间 $-1\leqslant x\leqslant 1$ 内的 $\cos(x)$,以及习题 2 中计算出的$P_4(x)$,$P_6(x)$和 $P_8(x)$。

   (b) 建立一个表,其列为在区间$[-1,1]$内等距的 19 个点 x 上的 $\cos(x)$,$P_4(x)$,

$P_6(x), P_8(x)$。

4.2 插值介绍

在第 4.1 节中我们了解了如何用泰勒多项式来逼近函数 $f(x)$，构造泰勒多项式所需的信息是 x_0 处的 f 及其导数值。该方法的缺点之一是必须知道函数的高阶导数值，而通常的情况是，它们要么无法得到，要么难以计算。

假设函数 $y=f(x)$ 在 $N+1$ 个点 $(x_0,y_0),\cdots,(x_N,y_N)$ 处的值已知，其中值 x_k 在区间 $[a,b]$ 上分布，且满足：

$$a \leqslant x_0 < x_1 < \cdots < x_N \leqslant b，且\ y_k = f(x_k)$$

则可以构造一个过这 $N+1$ 个点的 N 次多项式 $P(x)$。在构造中，只需知道 x_k 和 y_k 的数值，而不需要高阶导数值。可在整个区间 $[a,b]$ 内用多项式 $P(x)$ 来逼近 $f(x)$，然而，若需要求误差函数 $E(x)=f(x)-P(x)$，则需要知道 $f^{(N+1)}(x)$ 及其值的范围，即：

$$M = \max\{|f^{(N+1)}(x)| : a \leqslant x \leqslant b\}$$

统计和科学分析中经常出现函数 $y=f(x)$ 只在 $N+1$ 个点 (x_k,y_k) 处已知的情况，需要一种方法来近似求得 $f(x)$ 在其他点上的值。若已知值存在显著误差，则需考虑第 5 章中的曲线拟合方法。但是，若 (x_k,y_k) 已知具有高精度，则可以考虑通过这些点的多项式函数 $y=P(x)$。当 $x_0 < x < x_N$ 时，近似值 $P(x)$ 称为"内插值"，当 $x < x_0$ 或 $x_N < x$ 时，称 $P(x)$ 为"外插值"。在数值差分、数值积分以及绘制经过给定点的曲线的软件算法中，都有用多项式来计算函数近似值的情况。

下面简要地回顾一下如何计算多项式 $P(x)$：

$$P(x) = a_N x^N + a_{N-1}x^{N-1} + \cdots + a_2 x^2 + a_1 x + a_0 \tag{1}$$

霍纳方法是一种有效的计算方法。导数 $P'(x)$ 为：

$$P'(x) = N a_N x^{N-1} + (N-1)a_{N-1}x^{N-2} + \cdots + 2a_2 x^2 + a_1 \tag{2}$$

而满足 $I'(x)=P(x)$ 的不定积分 $I(x)=\int P(x)dx$ 为：

$$I(x) = \frac{a_N x^{N+1}}{N+1} + \frac{a_{N-1}x^N}{N} + \cdots + \frac{a_2 x^3}{3} + \frac{a_1 x^2}{2} + a_0 x + C \tag{3}$$

其中 C 为积分常数。算法 4.1(本节末尾)给出如何将霍纳方法用于 $P'(x)$ 和 $I(x)$ 的计算。

例 4.4 多项式 $P(x) = -0.02x^3 + 0.2x^2 - 0.4x + 1.28$ 通过 4 个点：(1,1.06)，(2,1.12)，(3,1.34)，(5,1.78)。计算：(a) $P(4)$，(b) $P'(4)$，(c) $\int_1^4 P(x)dx$，(d) $P(5.5)$，(e)给出如何计算 $P(x)$ 的系数。

对 $x=4$，利用算法 4.1(i)-(iii)（等价于表 1.2 中的过程）进行计算：

(a)

$$b_3 = a_3 = -0.02$$

$$b_2 = a_2 + b_3 x = 0.2 + (-0.02)(4) = 0.12$$

$$b_1 = a_1 + b_2 x = -0.4 + (0.12)(4) = 0.08$$

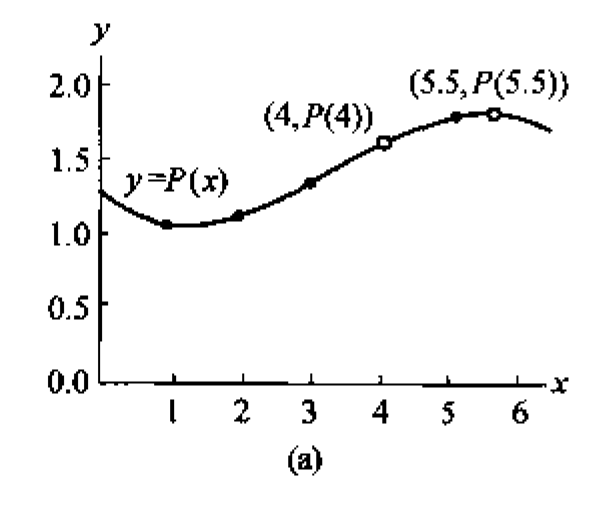
$$b_0 = a_0 + b_1 x = 1.28 + (0.08)(4) = 1.60$$

内插值为 $P(4)=1.60$（见图 4.7(a)）。

(b)
$$d_2 = 3a_3 = -0.06$$
$$d_1 = 2a_2 + d_2 x = 0.4 + (-0.06)(4) = 0.16$$
$$d_0 = a_1 + d_1 x = -0.4 + (0.16)(4) = 0.24$$

数值导数为 $P'(4) = 0.24$（见图 4.7(b)）。

(c)
$$i_4 = \frac{a_3}{4} = -0.005$$
$$i_3 = \frac{a_2}{3} + i_4 x = 0.06666667 + (-0.005)(4) = 0.04666667$$
$$i_2 = \frac{a_1}{2} + i_3 x = -0.2 + (0.04666667)(4) = -0.01333333$$
$$i_1 = a_0 + i_2 x = 1.28 + (-0.01333333)(4) = 1.22666667$$
$$i_0 = 0 + i_1 x = 0 + (1.22666667)(4) = 4.90666667$$



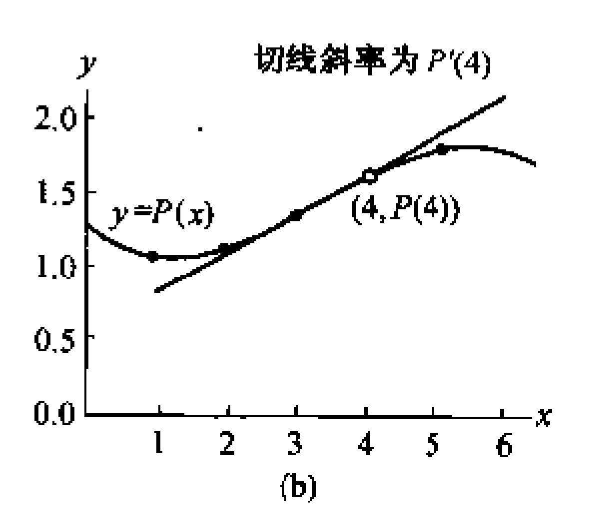


图 4.7 (a)逼近多项式 $P(x)$可用于内插点$(4, P(4))$和外插点$(5, P(5))$
(b)对逼近多项式 $P(x)$求导，且由 $P'(x)$计算内插点$(4, P(4))$的斜率

于是，$I(4) = 4.90666667$。类似地，$I(1) = 1.14166667$。因此，$\int_1^4 P(x)\,dx = I(4) - I(1) = 3.765$（见图 4.8）。

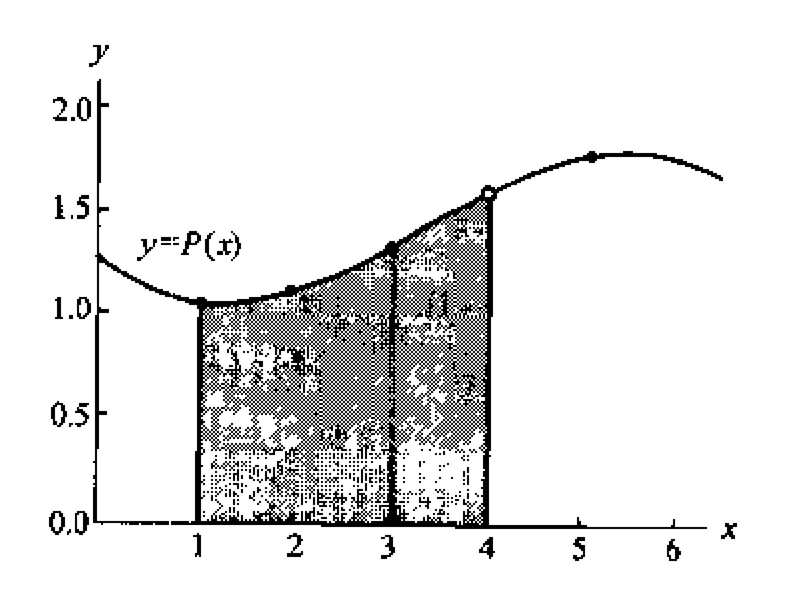


图 4.8 对逼近多项式 $P(x)$求积分，并用其不定积分计算区间 $1 \leqslant x \leqslant 4$ 内曲线下的面积

(d) 对 $x = 5.5$，利用算法 4.1(i)有：
$$b_3 = a_3 = -0.02$$
$$b_2 = a_2 + b_3 x = 0.2 + (-0.02)(5.5) = 0.09$$

$$b_1 = a_1 + b_2 x = -0.4 + (0.09)(5.5) = 0.095$$
$$b_0 = a_0 + b_1 x = 1.28 + (0.095)(5.5) = 1.8025$$

外插值为 $P(5.5) = 1.8025$（见图 4.7(a)）。

(e)可以用第 3 章的方法计算系数。设 $P(x) = A + Bx + Cx^2 + Dx^3$，则在每个点 $x = 1,2,3,5$ 处有关于 A, B, C, D 的线性方程：

$$\begin{aligned}
\text{At}x = 1: A + 1B + 1C + 1D &= 1.06 \\
\text{At}x = 2: A + 2B + 4C + 8D &= 1.12 \\
\text{At}x = 3: A + 3B + 9C + 27D &= 1.34 \\
\text{At}x = 5: A + 5B + 25C + 125D &= 1.78
\end{aligned} \tag{4}$$

式(4) 的解为：$A = 1.28,\ B = -0.4,\ C = 0.2,\ D = -0.2$。

用该方法来求解系数在数学上是可行的，但有时矩阵难以精确求解，本章设计了专门针对多项式计算的算法。

回到利用多项式计算一已知函数的近似值问题。在第 4.1 节中介绍过，$f(x) = \ln(1+x)$ 的 5 次泰勒多项式为：

$$T(x) = x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \frac{x^5}{5} \tag{5}$$

若在区间$[-1,1]$内用 $T(x)$来近似 $\ln(1+x)$，则在 $x = 0$ 处其误差为 0，而在 $x = 1$ 处（见表 4.4）误差最大；实际上，$T(1)$与正确值 $\ln(2)$之间的误差为 13%。要找到一个能在区间$[0,1]$内更好地逼近 $\ln(1+x)$的 5 次多项式。例 4.5 中的多项式 $P(x)$是一插值多项式，且在区间$[0,1]$内以不超过 0.00002385 的误差逼近 $\ln(1+x)$。

表 4.4　[0,1]内 5 次泰勒多项式 $T(x)$，以及函数 $\ln(1+x)$和误差 $\ln(1+x) - T(x)$的值

x	泰勒多项式 $T(x)$	函数 $\ln(1+x)$	误差 $\ln(1+x) - T(x)$
0.0	0.00000000	0.00000000	0.00000000
0.2	0.18233067	0.18232156	-0.00000911
0.4	0.33698133	0.33647224	-0.00050909
0.6	0.47515200	0.47000363	-0.00514837
0.8	0.61380267	0.58778666	-0.02601601
1.0	0.78333333	0.69314718	-0.09018615

例 4.5　考虑函数 $f(x) = \ln(1+x)$和基于 6 个节点 $x_k = k/5, k = 0,1,2,3,4,5$ 的多项式：

$$\begin{aligned}
P(x) = 0.02957206x^5 &- 0.12895295x^4 + 0.28249626x^3 \\
&- 0.48907554x^2 + 0.99910735x
\end{aligned}$$

下面是对近似 $P(x) \approx \ln(1+X)$的经验描述：

1. 在每个节点上有 $P(x_k) = f(x_k)$（见表 4.5）。

表 4.5　例 4.5 中的逼近多项式 $P(x)$和函数 $f(x) = \ln(1+x)$及误差 $E(x)$在$[-0.1,1.1]$内的值

x	逼近多项式 $P(x)$	函数 $f(x) = \ln(1+x)$	误差 $E(x) = f(x) - P(x)$
-0.1	-0.10509718	-0.10536052	-0.00026334
0.0	0.00000000	0.00000000	0.00000000

(续表)

x	逼近多项式 $P(x)$	函数 $f(x)=\ln(1+x)$	误差 $E(x)=f(x)-P(x)$
0.1	0.09528988	0.09531018	0.00002030
0.2	0.18232156	0.18232156	0.00000000
0.3	0.26237015	0.26236426	-0.00000589
0.4	0.33647224	0.33647224	0.00000000
0.5	0.40546139	0.40546511	0.00000372
0.6	0.47000363	0.47000363	0.00000000
0.7	0.53063292	0.53062825	-0.00000467
0.8	0.58778666	0.58778666	0.00000000
0.9	0.64184118	0.64185389	0.00001271
1.0	0.69314718	0.69314718	0.00000000
1.1	0.74206529	0.74193734	0.00012795

2. $[-0.1,1.1]$内最大误差在 $x=-0.1$ 处，且对 $-0.1\leqslant x\leqslant 1.1$ 有 $|误差|\leqslant 0.00026334$(见图 4.10)，因此曲线 $y=P(x)$与 $y=\ln(1+x)$几乎重合 (见图 4.9)。

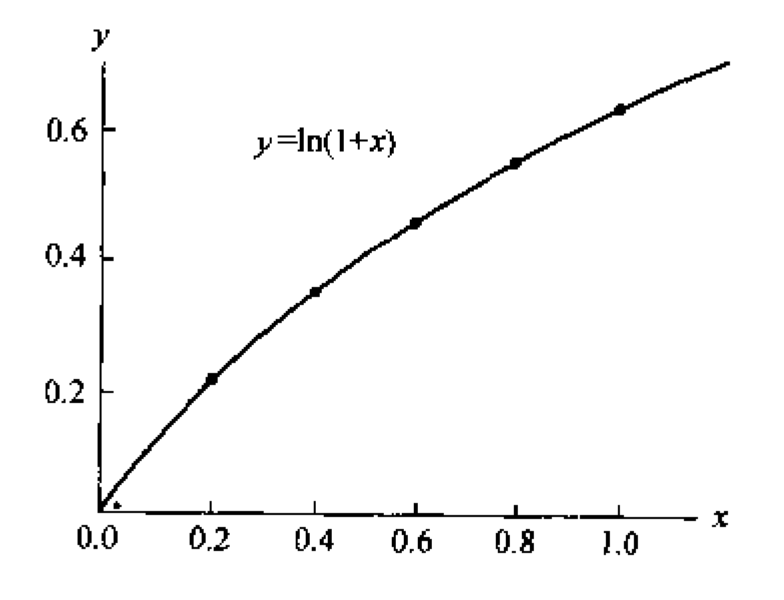


图 4.9 $y=P(x)$的曲线，它"在曲线 $y=\ln(1+x)$上方"

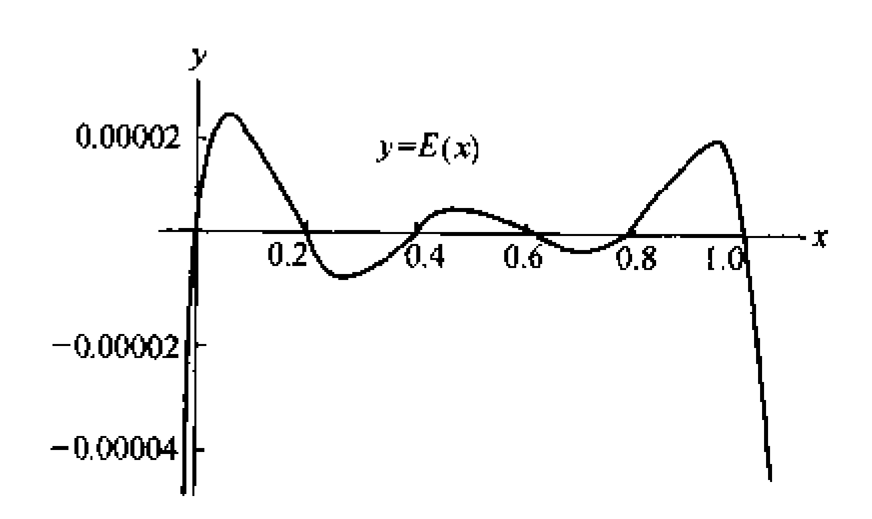


图 4.10 误差 $y=E(x)=\ln(1+x)-P(x)$的曲线

3. $[0,1]$内的最大误差在 $x=0.06472456$ 处，且对 $0\leqslant x\leqslant 1$ 有 $|误差|\leqslant 0.00002385$(见图 4.10)。

注：在节点 x_k 处有$f(x_k)=P(x_k)$，故在一节点处有 $E(x_k)=0$。$E(x)=f(x)-P(x)$看起来象一个振动的弹簧，其节点处是位移为 0 的横坐标。

算法 4.1(多项式计算) 用霍纳方法计算多项式 $P(x)$、其导数 $P'(x)$以及积分 $\int P(x)dx$的值

```
INPUT N                          {P(x)阶数}
INPUT A(0),A(1),…,A(N)           {P(x)系数}
INPUT C                          {积分常数}
INPUT X                          {独立变量}
```

(i) 计算 P(x)值 B(N): = A(N) FOR K = N - 1 DOWNTO 0 DO B(K): = A(K) + B(K+1) * X PRINT "The value P(x) is", B(0)	压缩版: Poly : = A(N) FOR K = N - 1 DOWNTO 0 DO Poly: = A(K) + Poly * X PRINT"The value P′(x)is", Poly
(ii) 计算 P′(x)值 D(N-1): = N * A(N) FOR K = N - 1 DOWNTO 1 DO D(K-1): = K * A(K) + D(K) * X PRINT"The value P′(x) is", D(0)	压缩版: Deriv: = N * A(N) FOR K = N - 1 DOWNTO 1 DO Deriv : = K * A(K) + Deriv * X PRINT "The value P′(x) is", Deriv
(iii) 计算 I(x)值 I(N+1): = A(N)/(N+1) FOR K: = N DOWNTO 1 DO I(K): = A(K-1)/K + I(K+1) * X I(0): C + I(1) * X PRINT "The value I(x) is" I(0)	压缩版: Integ: = A(N)/(N+1) FOR K = N DOWNTO 1 DO Integ : = A(K-1)/K + Integ * X Integ : = C + Integ * X PRINT "The value I(x) is", Integ

4.2.1 习题

1. 考虑经过 4 个点(1,1.54), (2,1.5), (3,1.42), (5,0.66)的函数 $P(x) = -0.02x^3 + 0.1x^2 - 0.2x + 1.66$。
   (a) 计算 $P(4)$。
   (b) 计算 $P'(4)$。
   (c) 计算 $P(x)$在区间[1,4]内的定积分。
   (d) 计算外插值 $P(5.5)$。
   (e) 给出计算 $P(x)$系数的方法。
2. 考虑经过 4 个点(0,2.08),(1,2.02),(2,2.00),(4,1.12)的函数 $P(x) = -0.04x^3 + 0.14x^2 - 0.16x + 2.08$。
   (a) 计算 $P(3)$。
   (b) 计算 $P'(3)$。
   (c) 计算 $P(x)$在区间[0,3]内的定积分。
   (d) 计算外插值 $P(4.5)$。
   (e) 给出计算 $P(x)$系数的方法。
3. 考虑经过 4 个点(1, 1.05), (2, 1.10), (3, 1.35), (5, 1.75)的函数 $P(x) = -0.0292166667x^3 + 0.275x^2 - 0.570833333x - 1.375$。
   (a) 证明:函数值 1.05,1.10,1.35,1.75 与例 4.4 中的值相差小于 1.8%,而 x^3 和 x 的系数相差大于 42%。
   (b) 计算 $P(4)$,并与例 4.4 相比较。
   (c) 计算 $P'(4)$,并与例 4.4 相比较。
   (d) 计算 $P(x)$在区间[1,4]内的定积分,并与例 4.4 相比较。

(e) 计算外插值 $P(5.5)$,并与例 4.4 相比较。

工工注:(a)部分表明,插值多项式系数的计算是一病态问题。

4.2.2 算法与程序

1. 用 MATLAB 写出实现算法 4.1 的程序,以 $1 \times N$ 矩阵 $P = [a_N \quad a_{N-1} \cdots a_2 \ a_1 \ a_0]$ 为多项式 $P(x) = a_N x^N + a_{N-1} x^{N-1} + \cdots + a_2 x^2 + a_1 x + a_0$ 的系数。
2. 对任意给定函数,过 6 个点 $(0,f(0))$,$(0.2,f(0.2))$,$(0.4,f(0.4))$,$(0.6,f(0.6)$,$(0.8,f(0.8))$,$(1,f(1))$ 的 5 次插值多项式 $P(x)$ 的 6 个系数为,其中:
$$P(x) = a_5 x^5 + a_4 x^4 + a_3 x^3 + a_2 x^2 + a_1 x + a_0$$
(i) 通过求解 6×6 线性方程组:
$$a_0 + a_1 x + a_2 x^2 + a_3 x^3 + a_4 x^4 + a_5 x^5 = f(x_j)$$
利用 $x_j = (j-1)/5, j = 1,2,3,4,5,6$,找出 $P(x)$ 的系数 $\{a_k\}_{k=0}^5$。
(ii) 利用问题 1 中的 MATLAB 程序计算内插值 $P(0.3)$, $P(0.4)$, $P(0.5)$,并分别与 $f(0.3)$,$f(0.4)$,$f(0.5)$ 比较。
(iii) 利用 MATLAB 程序计算外插值 $P(-0.1)$ 和 $P(1.1)$,并与 $f(-0.1)$ 和 $f(1.1)$ 比较。
(iv) 利用 MATLAB 程序计算 $P(x)$ 在$[0,1]$内的积分,并与 $f(x)$ 在$[0,1]$内的积分比较,在同一幅图中绘制$[0,1]$区间内 $f(x)$ 和 $P(x)$ 的曲线。
(v) 为 $P(x_k)$, $f(x_k)$ 和 $E(x_k) = f(x_k) - P(x_k)$ 制作一个表,其中 $x_k = k/100$,$k = 0,1,\cdots,100$。
(a) $f(x) = e^x$
(b) $f(x) = \sin(x)$
(c) $f(x) = (x+1)^{(x+1)}$
3. 一个游乐园的骑马路径采用 3 个多项式来建模。第 1 段为 1 次多项式 $P_1(x)$,覆盖一段在 110 英尺高度开始,在 60 英尺高度结束,水平距离为 100 英尺的路径;第 3 段也是一个 1 次多项式 $Q_1(x)$,覆盖一段起始高度为 65 英尺,终点高度为 70 英尺,水平距离为 50 英尺的路径。中间段应是一个多项式 $P(x)$(次数为最小可能),覆盖一段水平距离为 150 英尺的路径。
(a) 找出 $P(x)$, $P_1(x)$, $Q_1(x)$ 的表达式,使得 $P(100) = P_1(100)$, $P'(100) = P'_1(100)$, $P(250) = Q_1(250)$, $P(250) = Q_1(250)$, $P'(250) = Q'_1(250)$ 成立,并且 $P(x)$ 的曲率在 $x = 100$ 处与 $P_1(x)$ 相等,而在 $x = 250$ 处与 $Q_1(x)$ 相等。
(b) 在同一坐标系中画出 $P_1(x)$, $P(x)$, $Q_1(x)$ 的曲线。
(c) 利用算法 4.1(iii),找出给定水平距离上路径的平均高度。

4.3 拉格朗日逼近

插值就是利用相邻点上已知函数值的加权平均估计未知函数值。线性插值使用过两点的线段。(x_0, y_0) 和 (x_1, y_1) 之间的斜率为 $m = (y_1 - y_0)/(x_1 - x_0)$,直线 $y = m(x - x_0) + y_0$ 的

点 - 斜率公式可写为：

$$y = P(x) = y_0 + (y_1 - y_0)\frac{x - x_0}{x_1 - x_0} \tag{1}$$

式(1)展开的结果是一个次数$\leqslant 1$的多项式。在 x_0 和 x_1 处计算 $P(x)$的值得到 y_0 和 y_1：

$$P(x_0) = y_0 + (y_1 - y_0)(0) = y_0$$
$$P(x_1) = y_0 + (y_1 - y_0)(1) = y_1 \tag{2}$$

法国数学家约瑟夫·路易·拉格朗日使用了略微不同的方法，得出了该多项式。他注意到，它可以写成：

$$y = P_1(x) = y_0\frac{x - x_1}{x_0 - x_1} + y_1\frac{x - x_0}{x_1 - x_0} \tag{3}$$

(3)式的每一项包含一个线性因子，因此其和是一个次数$\leqslant 1$的多项式。(3)式中的商式用：

$$L_{1,0}(x) = \frac{x - x_1}{x_0 - x_1} \quad 和 \quad L_{1,1}(x) = \frac{x - x_0}{x_1 - x_0} \tag{4}$$

表示。易知 $L_{1,0}(x_0) = 1, L_{1,0}(x_1) = 0, L_{1,1}(x_0) = 0$ 和 $L_{1,1}(x_1) = 1$ 故(3)式中的多项式 $P_1(x)$ 也经过两给定点：

$$P_1(x_0) = y_0 + y_1(0) = y_0 \quad 和 \quad P_1(x_1) = y_0(0) + y_1 = y_1 \tag{5}$$

(4)式中的项 $L_{1,0}(x)$和 $L_{1,1}(x)$称为基于节点 x_0 和 x_1 的拉格朗日系数多项式。利用这种记号，(3)式可写为下面的求和表达式：

$$P_1(x) = \sum_{k=0}^{1} y_k L_{1,k}(x) \tag{6}$$

设纵坐标 y_k 由公式$y_k = f(x_k)$计算。若 $P_1(x)$用于在区间$[x_0, x_1]$内逼近$f(x)$，称该过程为内插；若 $x < x_0$，则利用 $x_1 < x$ 称为外插。下面的例子给出这些概念的示例。

例 4.6　考虑$[0.0, 1.2]$内的曲线 $y = f(x) = \cos(x)$。

(a) 利用节点 $x_0 = 0.0$ 和 $x_1 = 1.2$ 构造一线性插值多项式 $P_1(x)$。

(b) 利用节点 $x_0 = 0.2$ 和 $x_1 = 1.0$ 构造一线性插值多项式 $Q_1(x)$。

(a) 利用(3)式由横坐标 $x_0 = 5.00$ 和 $x_1 = 1.2$ 及纵坐标 $y_0 = \cos(0.0) = 1.000000$ 和 $y_1 = \cos(1.2) = 0.362358$ 得到：

$$\begin{aligned} P_1(x) &= 1.000000\frac{x - 1.2}{0.0 - 1.2} + 0.362358\frac{x - 0.0}{1.2 - 0.0} \\ &= -0.833333(x - 1.2) + 0.301965(x - 0.0) \end{aligned}$$

(b) 当使用节点 $x_0 = 0.2$ 和 $x_1 = 1.0$ 及 $y_0 = \cos(0.2) = 0.980067$ 和 $y_1 = \cos(1.0) = 0.540302$ 时，结果为：

$$\begin{aligned} Q_1(x) &= 0.980067\frac{x - 1.0}{0.2 - 1.0} + 0.540302\frac{x - 0.2}{1.0 - 0.2} \\ &= -1.225083(x - 1.0) + 0.675378(x - 0.2) \end{aligned}$$

图 4.11(a)和(b)显示 $y = \cos(x)$的图形，并将它分别与 $y = P_1(x)$和 $y = Q_1(x)$比较。其数值结果在表 4.6 中给出，可以看出，$Q_1(x)$对满足 x_k 的点 $0.1 \leqslant x_k \leqslant 1.1$ 有较小的误差；通过使用 $Q_1(x)$，列出的最大误差 $f(0.6) - P_1(0.6) = 0.144157$ 降至 $f(0.6) - Q_1(0.6) = 0.065151$。

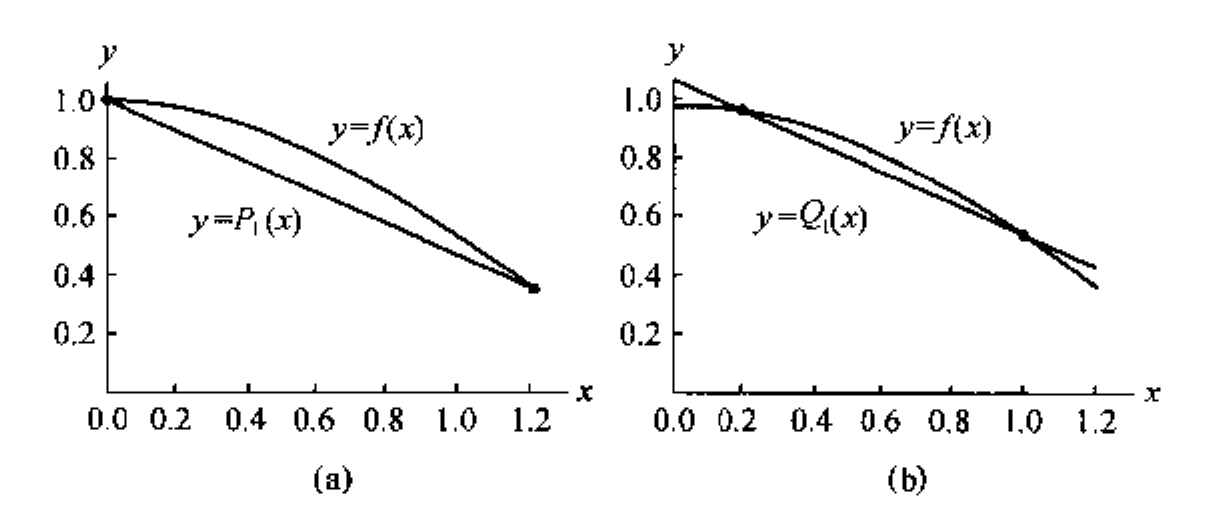


图 4.11 (a) 线性逼近 $y=P_1(x)$,其中节点 $x_0=0.0$ 和 $x_1=1.2$ 为区间$[a,b]$的端点
(b) 线性逼近 $y=Q_1(x)$,其中节点 $x_0=0.2$ 和 $x_1=1.0$ 在区间$[a,b]$内

表 4.6 $f(x)=\cos(x)$与线性逼近式 $P_1(x)$和 $Q_1(x)$的比较

x_k	$f(x_k)=\cos(x_k)$	$P_1(x_k)$	$f(x_k)-P_1(x_k)$	$Q_1(x_k)$	$f(x_k)-Q_1(x_k)$
0.0	1.000000	1.000000	0.000000	1.090008	-0.090008
0.1	0.995004	0.946863	0.048141	1.035037	-0.040033
0.2	0.980067	0.893726	0.086340	0.980067	0.000000
0.3	0.955336	0.840589	0.114747	0.925096	0.030240
0.4	0.921061	0.787453	0.133608	0.870126	0.050935
0.5	0.877583	0.734316	0.143267	0.815155	0.062428
0.6	0.825336	0.681179	0.144157	0.760184	0.065151
0.7	0.764842	0.628042	0.136800	0.705214	0.059628
0.8	0.696707	0.574905	0.121802	0.650243	0.046463
0.9	0.621610	0.521768	0.099842	0.595273	0.026337
1.0	0.540302	0.468631	0.071671	0.540302	0.000000
1.1	0.453596	0.415495	0.038102	0.485332	-0.031736
1.2	0.362358	0.362358	0.000000	0.430361	-0.068003

推广(6)式,得到构造经过 $N+1$ 个点$(x_0,y_0),(x_1,y_1),\cdots,(x_N,y_N)$的至多 N 次多项式 $P_N(x)$,形如:

$$P_N(x)=\sum_{k=0}^{N}y_kL_{N,k}(x) \tag{7}$$

其中 $L_{N,k}$为基于节点:

$$L_{N,k}(x)=\frac{(x-x_0)\cdots(x-x_{k-1})(x-x_{k+1})\cdots(x-x_N)}{(x_k-x_0)\cdots(x_k-x_{k-1})(x_k-x_{k+1})\cdots(x_k-x_N)} \tag{8}$$

的拉格朗日系数多项式。易知,项$(x-x_k)$和(x_k-x_k)不在式(8)的右端出现。可引入(8)式的乘式记号,写为:

$$L_{N,k}(x)=\frac{\prod_{\substack{j=0\\j\neq k}}^{N}(x-x_j)}{\prod_{\substack{j=0\\j\neq k}}^{N}(x_k-x_j)} \tag{9}$$

(9)中的记号说明在分子中有线性因子$(x-x_j)$,但不存在$(x-x_k)$。在分母中有类似的结构。

直接计算表明,对每个固定的 k,拉格朗日系数多项式 $L_{N,k}(x)$具有性质:

$$L_{N,k}(x_j)=1,当 j=k \quad 和 \quad L_{N,k}(x_j)=0,当 j\neq k \tag{10}$$

直接把这些值代入(7)式,可知多项式曲线 $y = P_N(x)$ 过点 (x_j, y_j):

$$\begin{aligned} P_N(x_j) &= y_0 L_{N,0}(x_j) + \cdots + y_j L_{N,j}(x_j) + \cdots + y_N L_{N,N}(x_j) \\ &= y_0(0) + \cdots + y_j(1) + \cdots + y_N(0) = y_j \end{aligned} \tag{11}$$

要证明 $P_N(x)$ 是惟一的,需要用到代数基本定理,一个次数小于等于 N 的多项式 $T(x)$ 至多有 N 个根。换言之,若 $T(x)$ 在横坐标上 $N+1$ 个不同点处为0,则它恒为0。设 $P_N(x)$ 不是惟一的,则存在另一个次数小于等于 N 的多项式 $Q_N(x)$ 也通过这 $N+1$ 个点。构造差多项式 $T(x) = P_N(x) - Q_N(x)$。可以注意到,$T(x)$ 次数小于等于 N,且对 $j = 0, 1, \ldots, N$, $T(x_j) = P_N(x_j) - Q_N(x_j) = y_j - y_j = 0$,故 $j = 0, 1, \ldots, N$,从而 $T(x) = 0$, $Q_N(x) = P_N(x)$。

(7)式展开的结果与(3)式类似。过3个点 (x_0, y_0), (x_1, y_1), (x_2, y_2) 的拉格朗日二次插值多项式为:

$$P_2(x) = y_0 \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)} + y_1 \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)} + y_2 \frac{(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)} \tag{12}$$

过4个点 (x_1, y_0), (x_1, y_1), (x_2, y_2), (x_3, y_3) 的拉格朗日二次插值多项式为

$$\begin{aligned} P_3(x) = & y_0 \frac{(x - x_1)(x - x_2)(x - x_3)}{(x_0 - x_1)(x_0 - x_2)(x_0 - x_3)} + y_1 \frac{(x - x_0)(x - x_2)(x - x_3)}{(x_1 - x_0)(x_1 - x_2)(x_1 - x_3)} \\ & + y_2 \frac{(x - x_0)(x - x_1)(x - x_3)}{(x_2 - x_0)(x_2 - x_1)(x_2 - x_3)} + y_3 \frac{(x - x_0)(x - x_1)(x - x_2)}{(x_3 - x_0)(x_3 - x_1)(x_3 - x_2)} \end{aligned} \tag{13}$$

例4.7 考虑[0.0,1.2]内的函数 $y = f(x) = \cos(x)$。

(a) 用3个节点 $x_0 = 0.0$, $x_1 = 0.6$ 和 $x_2 = 1.2$ 构造一个二次插值多项式 $P_2(x)$。

(b) 用4个节点 $x_0 = 0.0$, $x_1 = 0.4$, $x_2 = 0.8$ 和 $x_3 = 1.2$ 构造一个三次插值多项式 $P_3(x)$。

在式(12)中使用 $x_0 = 0.0$, $x_1 = 0.6$, $x_2 = 1.2$ 和 $y_0 = \cos(0.0) = 1$, $y_1 = \cos(0.6) = 0.825336$, $y_2 = \cos(1.2) = 0.362358$,得:

$$\begin{aligned} P_2(x) = & 1.0 \frac{(x - 0.6)(x - 1.2)}{(0.0 - 0.6)(0.0 - 1.2)} + 0.825336 \frac{(x - 0.0)(x - 1.2)}{(0.6 - 0.0)(0.6 - 1.2)} \\ & + 0.362358 \frac{(x - 0.0)(x - 0.6)}{(1.2 - 0.0)(1.2 - 0.6)} \\ = & 1.388889(x - 0.6)(x - 1.2) - 2.292599(x - 0.0)(x - 1.2) \\ & + 0.503275(x - 0.0)(x - 0.6) \end{aligned}$$

在式(13)中利用 $x_0 = 0.0$, $x_1 = 0.4$, $x_2 = 0.8$, $x_3 = 1.2$ 和 $y_0 = \cos(0.0) = 1.0$, $y_1 = \cos(0.4) = 0.921061$, $y_2 = \cos(0.8) = 0.696707$ 和 $y_3 = \cos(1.2) = 0.362358$,得:

$$\begin{aligned} P_3(x) = & 1.000000 \frac{(x - 0.4)(x - 0.8)(x - 1.2)}{(0.0 - 0.4)(0.0 - 0.8)(0.0 - 1.2)} \\ & + 0.921061 \frac{(x - 0.0)(x - 0.8)(x - 1.2)}{(0.4 - 0.0)(0.4 - 0.8)(0.4 - 1.2)} \\ & + 0.696707 \frac{(x - 0.0)(x - 0.4)(x - 1.2)}{(0.8 - 0.0)(0.8 - 0.4)(0.8 - 1.2)} \\ & + 0.362358 \frac{(x - 0.0)(x - 0.4)(x - 0.8)}{(1.2 - 0.0)(1.2 - 0.4)(1.2 - 0.8)} \\ = & -2.604167(x - 0.4)(x - 0.8)(x - 1.2) \end{aligned}$$

$$+ 7.195789(x - 0.0)(x - 0.8)(x - 1.2)$$
$$- 5.443021(x - 0.0)(x - 0.4)(x - 1.2)$$
$$+ 0.943641(x - 0.0)(x - 0.4)(x - 0.8)$$

$y = \cos(x)$和多项式 $y = P_2(x)$及 $y = P_3(x)$的曲线分别在图 4.12(a) 和图 4.12(b) 中给出。

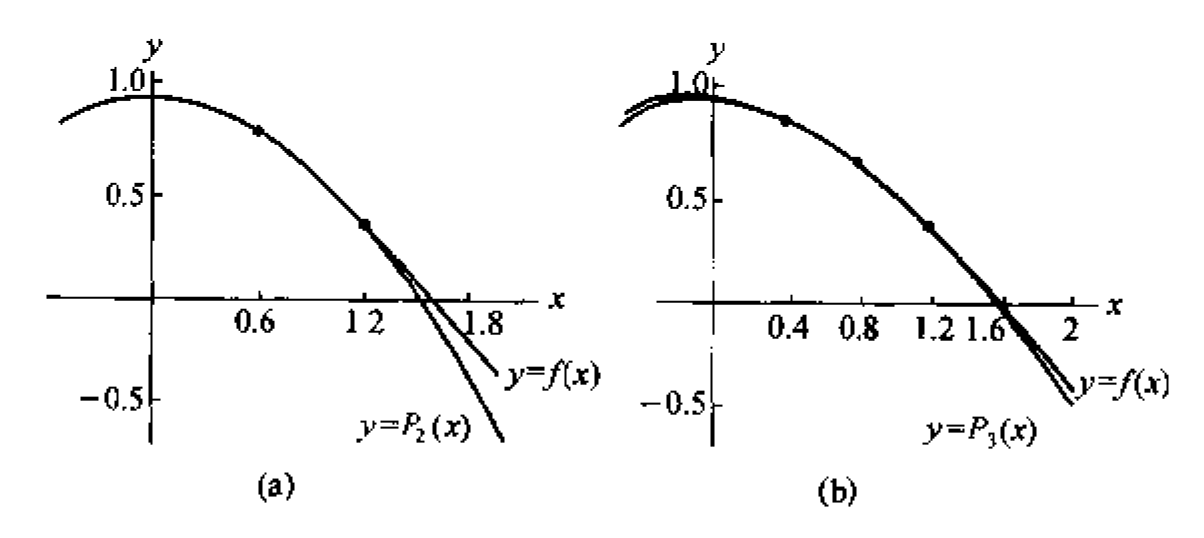


图 4.12 (a) 基于节点 $x_0 = 0.0, x_1 = 0.6$ 和 $x_2 = 1.2$ 的二次逼近多项式 $y = P_2(x)$; (b) 基于节点 $y = P_3(x), x_0 = 0.0, x_1 = 0.4, x_2 = 0.8$ 和 $x_3 = 1.2$ 的立方逼近多项式 $y = P_3(x)$

4.3.1 误差项和误差界

当利用拉格朗日多项式来逼近一连续函数 $f(x)$时,了解其误差项的属性非常重要。它与泰勒多项式的误差项相似,只是用乘积$(x - x_0)(x - x_1)\cdots(x - x_N)$替换了因子$(x - x_0)^{N+1}$。这与预期相符,因为插值在 $N + 1$ 个节点 x_k 上进行,在这些节点上有 $E_N(x_k) = f(x_k) - P_N(x_k) = y_k - y_k = 0, k = 0,1,2,\cdots,N$。

定理 4.3(拉格朗日多项式逼近) 设$f \in C^{N+1}[a,b]$,且 $x_0, x_1, \cdots, x_N \in [a,b]$为 $N + 1$ 个节点。若 $x \in [a,b]$,则:

$$f(x) = P_N(x) + E_N(x) \tag{14}$$

其中 $P_N(x)$为一多项式,可用于近似$f(x)$:

$$f(x) \approx P_N(x) = \sum_{k=0}^{N} f(x_k) L_{N,k}(x) \tag{15}$$

误差项 $E_N(x)$形如:

$$E_N(x) = \frac{(x - x_0)(x - x_1)\cdots(x - x_N) f^{(N+1)}(c)}{(N + 1)!} \tag{16}$$

对区间$[a,b]$内的某个值 c。

证明:作为一个一般方法的例子,我们建立 $N = 1$ 时的式(16),一般情况在练习中讨论。从定义函数 $g(t)$开始:

$$g(t) = f(t) - P_1(t) - E_1(x)\frac{(t - x_0)(t - x_1)}{(x - x_0)(x - x_1)} \tag{17}$$

注意:x, x_0, x_1 都是与变量 t 相关的常数,且在这 3 个点上 $g(t)$的值为 0,即:

$$g(x) = f(x) - P_1(x) - E_1(x)\frac{(x - x_0)(x - x_1)}{(x - x_0)(x - x_1)} = f(x) - P_1(x) - E_1(x) = 0$$

$$g(x_0)=f(x_0)-P_1(x_0)-E_1(x)\frac{(x_0-x_0)(x_0-x_1)}{(x-x_0)(x-x_1)}=f(x_0)-P_1(x_0)=0$$

$$g(x_1)=f(x_1)-P_1(x_1)-E_1(x)\frac{(x_1-x_0)(x_1-x_1)}{(x-x_0)(x-x_1)}=f(x_1)-P_1(x_1)=0$$

设 x 在开区间(x_0,x_1)内，在区间$[x_0,x]$内对 $g(t)$利用罗尔定理可找到一个值 d_0，满足 $x_0<d_0<x$：

$$g'(d_0)=0 \tag{18}$$

在区间$[x,x_1]$再次对 $g(t)$应用罗尔定理，可找到一个值 d_1，$x<d_1<x_1$，满足：

$$g'(d_1)=0 \tag{19}$$

式(18)和式(19)说明函数 $g'(t)$在 $t=d_0$ 和 $t=d_1$ 处为 0，对 $g'(t)$在区间$[d_0,d_1]$内第三次应用罗尔定理，得到值 c，有：

$$g^{(2)}(c)=0 \tag{20}$$

回到式(17)计算导数 $g'(t)$和 $g''(t)$：

$$g'(t)=f'(t)-P'_1(t)-E_1(x)\frac{(t-x_0)+(t-x_1)}{(x-x_0)(x-x_1)} \tag{21}$$

$$g''(t)=f''(t)-0-E_1(x)\frac{2}{(x-x_0)(x-x_1)} \tag{22}$$

在式(22)中，由于 $P_1(t)$是次数 $N=1$ 的多项式，故其 2 阶导数 $P''_1(t)\equiv 0$，使用(20)在点 $t=c$ 处计算式(22)得：

$$0=f''(c)-E_1(x)\frac{2}{(x-x_0)(x-x_1)} \tag{23}$$

由式(23)解得 $E_1(x)$为形如式(16)的余项：

$$E_1(x)=\frac{(x-x_0)(x-x_1)f^{(2)}(c)}{2!} \tag{24}$$

证明完毕。

下面的结果说明了在拉格朗日多项式的节点为等距的 $x_k=x_0+hk$，$k=0,1,\cdots,N$ 时的特殊情况，该多项式 $P_N(x)$只能用于求区间$[x_0,x_N]$内的插值。

定理 4.4（等距节点拉格朗日多项式的误差界） 设 $f(x)$定义在$[a,b]$内，等距节点 $x_k=x_0+hk$ 在该区间内，并设 $f(x)$和直到 $N+1$ 阶导数在子区间$[x_0,x_1]$，$[x_0,x_2]$和$[x_0,x_3]$内连续且有界，即，对 $N=1,2,3$：

$$|f^{(N+1)}(x)|\leqslant M_{N+1},\quad x_0\leqslant x\leqslant x_N \tag{25}$$

对应于 $N=1,2,3$，误差项(16) 具有如下的界：

$$|E_1(x)|\leqslant\frac{h^2M^2}{8},\quad x\in[x_0,x_1] \tag{26}$$

$$|E_2(x)|\leqslant\frac{h^3M^3}{9\sqrt{3}},\quad x\in[x_0,x_2] \tag{27}$$

$$|E_3(x)|\leqslant\frac{h^4M_4}{24},\quad x\in[x_0,x_3] \tag{28}$$

证明:只证式(26),其余留给读者自己证明。利用变量替换 $x - x_0 = t$ 和 $x - x_1 = t - h$,误差项 $E_1(x)$ 可写作:

$$E_1(x) = E_1(x_0 + t) = \frac{(t^2 - ht)f^{(2)}(c)}{2!} \quad , \quad 0 \leqslant t \leqslant h \tag{29}$$

其中导数的界为:

$$|f^{(2)}(c)| \leqslant M_2 \quad , \quad x_0 \leqslant c \leqslant x_1 \tag{30}$$

现在来确定式(29)分子中 $(t^2 - ht)$ 的界,称该项为 $\Phi(t) = t^2 - ht$。由于 $\Phi'(t) = 2t - h$,故存在一个临界点 $t = h/2$ 为 $\Phi'(t) = 0$ 的解。$\Phi(t)$ 在 $[0, h]$ 内的极值在端点 $\Phi(0) = 0$, $\Phi(h) = 0$,或临界点 $\Phi(h/2) = -h^2/4$ 处得到。由于后者的值最大,故可得:

$$|\Phi(t)| = |t^2 - ht| \leqslant \frac{|-h^2|}{4} = \frac{h^2}{4} \quad , \quad 0 \leqslant t \leqslant h \tag{31}$$

利用式(30)和式(31)来估计式(29)分子中的乘积,得:

$$|E_1(x)| = \frac{|\Phi(t)||f^{(2)}(c)|}{2!} \leqslant \frac{h^2 M_2}{8} \tag{32}$$

从而式(26)得证。

4.3.2 比较精度与 $O(h^{N+1})$

定理 4.4 的重要性在于了解线性、二次和三次插值的误差项大小之间的简单关系。在每一种情况中,误差界 $|E_N(x)|$ 在两个方面依赖于 h:第一,h^{N+1} 是显式的,故 $|E_N(x)|$ 正比于 h^{N+1};第二,值 M_{N+1} 通常依赖于 h,且随着 h 趋近于 0 而趋近于 $|f^{(N+1)}(x_0)|$。因此,当 h 趋近于 0 时,$|E_N(x)|$ 收敛于 0 的速度与 h^{N+1} 收敛于 0 的速度相同。在讨论这一特点时用到记号 $\boldsymbol{O}(h^{N+1})$,例如,式(26)的误差界可表示为:

$$|E_1(x)| = \boldsymbol{O}(h^2) \quad , \quad x \in [x_0, x_1]$$

用记号 $\boldsymbol{O}(h^2)$ 代替式式(26)中的 $h^2 M_2/8$,表示误差项的界近似为 h^2 的倍数。即:

$$|E_1(x)| \leqslant Ch^2 \approx \boldsymbol{O}(\mathrm{h}^2)$$

结果是,若 $f(x)$ 的导数在区间 $[a, b]$ 内一致有界,且 $|h| < 1$。则选择大的 N 将得到小的 h^{N+1},从而高次逼近多项式将产生较小的误差。

例 4.8 考虑 $[0.0, 1.2]$ 内的 $y = f(x) = \cos(x)$。利用式(26)~式(28)公式来确定例 4.6 和例 4.7 中的拉格朗日多项式 $P_1(x)$, $P_2(x)$ 和 $P_3(x)$ 的误差界。

首先确定导数 M_2, M_3 和 M_4 在区间 $[0.0, 1.2]$ 内的界 $|f^{(2)}(x)|$, $|f^{(3)}(x)|$ 和 $|f^{(4)}(x)|$:

$$|f^{(2)}(x)| = |-\cos(x)| \leqslant |-\cos(0.0)| = 1.000000 = M_2$$

$$|f^{(3)}x| = |\sin(x)| \leqslant |\sin(1.2)| = 0.932039 = M_3$$

$$|f^{(4)}(x)| = |\cos(x)| \leqslant |\cos(0.0)| = 1.000000 = M_4$$

对 $P_1(x)$,节点的间距为 $h = 1.2$,其误差界为:

$$|E_1(x)| \leqslant \frac{h^2 M^2}{8} \leqslant \frac{(1.2)^2(1.000000)}{8} = 0.180000 \tag{33}$$

对 $P_2(x)$,节点间距为 $h = 0.6$,其误差界为:

$$|E_2(x)| \leqslant \frac{h^3 M_3}{9\sqrt{3}} \leqslant \frac{(0.6)^3(0.932039)}{9\sqrt{3}} = 0.012915 \tag{34}$$

对 $P_3(x)$，节点间距为 $h=0.4$，其误差界为：

$$|E_3(x)| \leqslant \frac{h^4 M_4}{24} \leqslant \frac{(0.4)^4(1.000000)}{24} = 0.001067 \tag{35}$$

从例 4.6 中可以看出，$|E_1(0.6)| = |\cos(0.6) - P_1(0.6)| = 0.144157$，故式(33)中的界 0.180000 是合理的。图 4.13(a) 和图 4.13(b) 分别显示了误差函数 $E_1(x) = \cos(x) - P_2(x)$ 和 $E_3(x) = \cos(x) - P_3(x)$，其数值计算在表 4.7 中给出。利用表中的值可以得到 $|E_2(1.0)| = |\cos(1.0) - P_2(1.0)| = 0.008416$ 和 $|E_3(0.2)| = |\cos(0.2) - P_3(0.2)| = 0.000855$，与式(34)和式(35)中的界 0.012915 和 0.001607 一致。

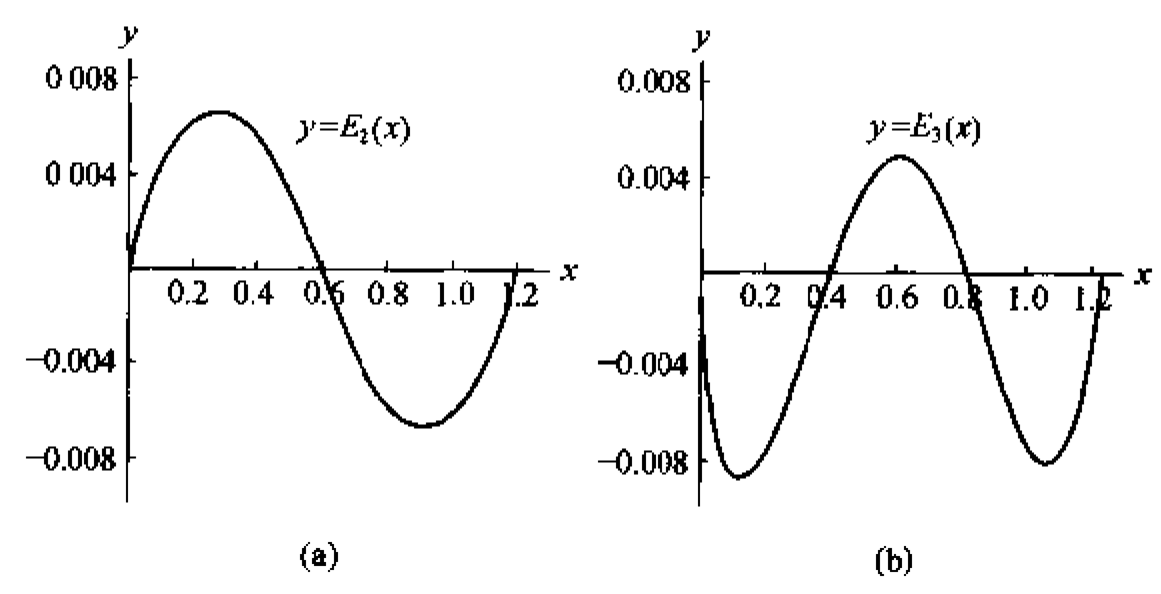


图 4.13　(a) 误差函数 $E_2(x) = \cos(x) - P_2(x)$；(b) 误差函数 $E_3(x) = \cos(x) - P_3(x)$

表 4.7　$f(x) = \cos(x)$ 及二次和三次逼近多项式 $P_2(x)$ 和 $P_3(x)$ 的比较

x_k	$f(x_k) = \cos(x_k)$	$P_2(x_k)$	$E_3(x_k)$	$P_2(x_k)$	$E_2(x_k)$
0.0	1.000000	1.000000	0.0	1.000000	0.0
0.1	0.995004	0.990911	0.004093	0.995835	−0.000831
0.2	0.980067	0.973813	0.006253	0.980921	−0.000855
0.3	0.955336	0.948707	0.006629	0.955812	−0.000476
0.4	0.921061	0.915592	0.005469	0.921061	0.0
0.5	0.877583	0.874468	0.003114	0.877221	0.000361
0.6	0.825336	0.825336	0.0	0.824847	0.00089
0.7	0.764842	0.768194	−0.003352	0.764491	0.000351
0.8	0.696707	0.703044	−0.006338	0.696707	0.0
0.9	0.621610	0.629886	−0.008276	0.622048	−0.000438
1.0	0.540302	0.548719	−0.008416	0.541068	−0.000765
1.1	0.453596	0.459542	−0.005946	0.454320	−0.000274
1.2	0.362358	0.362358	0.0	0.362358	0.0

4.3.3 MATLAB

下面的程序通过构造各项为拉格朗日多项式系数的向量来找出经过给定点的组合多项式(collocation polynomial)。程序使用了命令 poly 和 conv。poly 命令创建一个向量,其项为一多项式的系数,该多项式具有给定的根。conv 命令生成一个向量,其项为多项式系数,该多项式是另外两个多项式的积。

例 4.9 找出两个 1 次多项式 $P(x)$和 $Q(x)$的积,它们的根分别为 2 和 3。

```
>> P = poly(2)
P =
   1  -2
>> Q = poly(3)
Q =
   1  -3
>> conv(P,Q)
ans =
   1  -5  6
```

故,$P(x)$与 $Q(x)$的乘积为 x^2-5x+6。

程序 4.1(拉格朗日逼近) 基于 $N+1$ 个点 $P(x)=\sum_{k=0}^{N} y_k L_{N,k}(x)$ $k=0,1,\cdots,N$ 计算拉格朗日多项式(x_k, y_k)

```
function [C,L] = lagran(X,Y)
% Input   - X is a vector that contains a list of abscissas
%         - Y is a vector that contains a list of ordinates
% Output  - C is a matrix that contains the coefficients of
%           the Lagrange interpolatory polynomial
%         - L is a matrix that contains the Lagrange
%           coefficient folynomials
w = length(X);
n = w - 1;
L = zeros(w,w);
% Form the Lagrange coefficient polynomials
for k = 1:n+1
    V = 1;
    for j = 1:n+1
    if k~ = j
    V = conv(V,poly(X(j)))/(X(k) - X(j));
    end
end
    L(k,:) = V;
end
% Determine the coefficients of the Lagrange interpolating
% polynomial
C = Y * L;
```

4.3.4 习题

1. 找出逼近 $f(x)=x^3$ 的拉格朗日多项式。
   (a) 利用节点 $x_0=-1$ 和 $x_1=0$ 求线性插值多项式 $P_1(x)$。
   (b) 利用节点 $x_0=-1, x_1=0$ 和 $x_2=1$,求二次插值多项式 $P_2(x)$。
   (c) 利用节点 $x_0=-1, x_1=0, x_2=1$ 和 $x_3=2$ 求三次逼近多项式 $P_3(x)$。
   (d) 利用节点 $x_0=1, x_1=2$,求线性插值多项式 $P_1(x)$。
   (e) 利用节点 $x_0=0, x_1=1$ 和 $x_2=2$,求二次插值多项式 $P_2(x)$。
2. 设 $f(x)=x+2/x$
   (a) 用基于点 $x_0=1, x_1=2$ 和 $x_2=2.5$ 的二次拉格朗日多项式,求 $f(1.5)$ 和 $f(1.2)$ 的近似值。
   (b) 用基于点 $x_0=0.5, x_1=1, x_2=2$ 和 $x_3=2.5$ 的三次拉格朗日多项式,求 $f(1.5)$ 和 $f(1.2)$ 的近似值。
3. 设 $f(x)=2\sin(\pi x/6)$,其中 x 为弧度
   (a) 用基于点 $x_0=0, x_1=1$ 和 $x_2=3$ 的二次拉格朗日插值求 $f(2)$ 和 $f(2.4)$ 的近似值。
   (b) 用基于点 $x_0=0, x_1=1, x_2=3$ 和 $x_3=5$ 的三次拉格朗日插值求 $f(2)$ 和 $f(2.4)$ 的近似值。
4. 设 $f(x)=2\sin(\pi x/6)$,其中 x 为弧度
   (a) 用基于点 $x_0=0, x_1=1$ 和 $x_2=3$ 的二次拉格朗日插值求 $f(4)$ 和 $f(3.5)$ 的近似值。
   (b) 用基于点 $x_0=0, x_1=1, x_2=3$ 和 $x_3=5$ 的三次拉格朗日插值求 $f(4)$ 和 $f(3.5)$ 的近似值。
5. 写出 $f(x)$ 的3次拉格朗日插值多项式的误差项 $E_3(x)$,插值节点为 $x_0=-1, x_1=0, x_2=3$ 和 $x_4=4$,而 $f(x)$ 为:
   (a) $f(x)=4x^3-3x+2$
   (b) $f(x)=x^4-2x^3$
   (c) $f(x)=x^5-5x^4$
6. 设 $f(x)=x^x$
   (a) 求节点为 $x_0=1, x_1=1.25$ 和 $x_2=1.5$,时的2次拉格朗日多项式 $P_2(x)$ 式。
   (b) 用(a)中的多项式估计 $f(x)$ 在区间[1,1.5]内的平均值。
   (c) 利用定理4.4中的式(27),求用 $P_2(x)$ 近似 $f(x)$ 的误差界。
7. 考虑节点为 x_0, x_1 和 x_2 的2次拉格朗日多项式的系数多项式 $L_{2,k}(x)$,定义 $g(x)=L_{2,0}(x)+L_{2,1}(x)+L_{2,2}(x)-1$。
   (a) 证明 g 为次数小于等于2的多项式。
   (b) 证明对 $k=0,1,2, g(x_k)=0$。
   (c) 证明对任意 $x, g(x)=0$。提示:利用代数基本定理。
8. 设 $L_{N,0}(x), L_{N,1}(x), \cdots, L_{N,N}(x)$ 是节点为 $N+1$ 个点 $x_0, x_1, \cdots, x_N$ 的拉格朗日多项式的系数多项式,证明对任意实数 x, $\sum_{k=0}^{N} L_{N,k}(x)=1$。

9. 设$f(x)$为次数小于等于N的多项式,设$P_N(x)$为基于$N+1$个节点$x_0, x_1, \cdots, x_N$的次数小于等于N的拉格朗日多项式。证明对所有x,$f(x)=P_N(x)$。提示:证明误差项$E_N(x)$恒为0。
10. 考虑区间[0,1]内的函数$f(x)=\sin(x)$,利用定理4.4来确定步长h,使得:
   (a) 线性拉格朗日插值的精度为10^{-6}(即,求h使得$|E_1(x)|<5\times 10^{-7}$)。
   (b) 2次拉格朗日插值的精度为10^{-6}(即,求h使得$|E_2(x)|<5\times 10^{-7}$)。
   (c) 3次拉格朗日插值的精度为10^{-6}(即,求h使得$|E_3(x)|<5\times 10^{-7}$)。
11. 由式(16)和$N=2$证明不等式(27)。设$x_1=x_0+h$,$x_2=x_0+2h$,证明:若$x_0\leqslant x\leqslant x_2$,则:

$$|x-x_0||x-x_1||x-x_2|\leqslant \frac{2h^3}{3\times 3^{1/2}}$$

   提示:在区间$-h\leqslant t\leqslant h$内,利用变量替换$t=x-x_1$,$t+h=x-x_0$和$t-h=x-x_2$,以及函数$v(t)=t^3-th^2$。令$v'(t)=0$并求解t为h的函数。
12. 二维线性插值。考虑过3个点(x_0,y_0,z_0),(x_1,y_1,z_1)和(x_2,y_2,z_2)的多项式$z=P(x,y)=A+Bx+Cy$,则A,B,C为线性方程组的解:

$$\begin{aligned} A + Bx_0 + Cy_0 &= z_0 \\ A + Bx_1 + Cy_1 &= z_1 \\ A + Bx_2 + Cy_2 &= z_2 \end{aligned}$$

   (a) 求A,B,C,使得$z=P(x,y)$过点(1,1,5),(2,1,3)和(1,2,9)。
   (b) 求A,B,C,使得$z=P(x,y)$过点(1,1,2.5),(2,1,0)和(1,2,4)。
   (c) 求A,B,C,使得$z=P(x,y)$过点(2,1,5),(1,3,7)和(3,2,4)。
   (d) 能否找到值A,B,C,使得$z=P(x,y)$过点(1,2,5),(3,2,7)和(1,2,0)?为什么?
13. 利用定理1.7,广义罗尔定理和函数:

$$g(t)=f(t)-P_N(t)-E_n(x)\frac{(t-x_0)(t-x_1)\cdots(t-x_N)}{(x_1-x_0)(x-x_1)\cdots(x-x_N)}$$

   其中$P_N(x)$为N次拉格朗日多项式,证明:误差项$E_N(x)=f(x)-P_N(x)$具有形式:

$$E_N(x)=(x-x_0)(x-x_1)\cdots(x-x_N)\frac{f^{(N+1)}(c)}{(N+1)!}$$

   提示:找出$g^{(N+1)}(t)$,然后在$t=c$处求其值。

4.3.5 算法与程序

1. 利用程序4.1,求第4.2节的算法与程序中问题2(i)a,b,c中插值多项式的系数。在同一坐标系中画出每个函数和相应插值多项式的曲线。
2. 下表给出11月8日洛杉矶的某个郊区在5个小时中的测量温度。
   (a) 利用程序4.1,对表中的数据构造一个拉格朗日插值多项式。
   (b) 利用算法4.1(iii)估计在这5小时内的平均温度。
   (c) 在同一坐标系中画出表中的数据和由(a)得到的多项式。讨论用(a)中的多项式计算平均温度可能产生的误差。

下午时间	华氏温度
1	66
2	66
3	65
4	64
5	63
6	63

4.4　牛顿多项式

有时需要找出若干逼近多项式 $P_1(x), P_2(x), \cdots, P_N(x)$ 然后从中选择最适合的。若用拉格朗日多项式，则在 $P_{N-1}(x)$ 和 $P_N(x)$ 之间没有构造上的联系，每个多项式需要单独构造，而且计算高次多项式需要大量的工作。我们采用一种新的方法来构造牛顿多项式，它们具有递归关系：

$$P_1(x) = a_0 + a_1(x - x_0) \tag{1}$$

$$P_2(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) \tag{2}$$

$$P_3(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + a_3(x - x_0)(x - x_1)(x - x_2) \tag{3}$$

$$\vdots$$

$$P_N(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + a_3(x - x_0)(x - x_1)(x - x_2) + a_4(x - x_0)(x - x_1)(x - x_2)(x - x_3) + \cdots + a_N(x - x_0)\cdots(x - x_{N-1}) \tag{4}$$

这里多项式 $P_N(x)$ 可由 $P_{N-1}(x)$ 通过递归关系得到。

$$P_N(x) = P_{N-1}(x) + a_N(x - x_0)(x - x_1)(x - x_2)\cdots(x - x_{N-1}) \tag{5}$$

多项式(4)称为具有 N 个中心 $x_0, x_1, \cdots, x_{N-1}$ 的牛顿多项式，它是线性因子乘积的和，其中的最高次项为：

$$a_N(x - x_0)(x - x_1)(x - x_2)\cdots(x - x_{N-1})$$

因此 $P_N(x)$ 是一个次数小于等于 N 的普通多项式。

例 4.10　给定中心 $x_0 = 1, x_1 = 3, x_2 = 4$ 和 $x_3 = 4.5$ 及系数 $a_0 = 5, a_1 = -2, a_2 = 0.5, a_3 = -0.1$ 和 $a_4 = 0.003$，求 $P_1(x), P_2(x), P_3(x)$ 和 $P_4(x)$ 并对 $k = 1,2,3,4$ 计算 $P_k(2.5)$。

利用式(1)～(4)，有：

$$P_1(x) = 5 - 2(x - 1),$$

$$P_2(x) = 5 - 2(x - 1) + 0.5(x - 1)(x - 3),$$

$$P_3(x) = P_2(x) - 0.1(x - 1)(x - 3)(x - 4),$$

$$P_4(x) = P_3(x) + 0.003(x - 1)(x - 3)(x - 4)(x - 4.5)$$

在 $x = 2.5$ 处计算多项式的值，得到：

$$P_1(2.5) = 5 - 2(1.5) = 2$$

$$P_2(2.5) = P_1(2.5) + 0.5(1.5)(-0.5) = 1.625$$

$$P_3(2.5) = P_2(2.5) - 0.1(1.5)(-0.5)(-1.5) = 1.5125$$
$$P_4(2.5) = P_3(2.5) + 0.003(1.5)(-0.5)(-1.5)(-2.0) = 1.50575$$

4.4.1 嵌套乘法

若 N 固定且多次计算多项式x_k 的值,则应使用嵌套乘法。该过程与一般多项式的嵌套乘法类似,只是必须从独立变量 x 中将中心 x_k 减去。$P_N(x)$的嵌套乘法形式为:

$$P_3(x) = ((a_3(x - x_2) + a_2)(x - x_1) + a_1)(x - x_0) + a_0 \tag{6}$$

要对给定 x 值计算 $P_3(x)$,从最内层开始,逐步地得到值:

$$\begin{aligned} S_3 &= a_3 \\ S_2 &= S_3(x - x_2) + a_2 \\ S_1 &= S_2(x - x_1) + a_1 \\ S_0 &= S_1(x - x_0) + a_0 \end{aligned} \tag{7}$$

值 S_0 即为 $P_3(x)$。

例 4.11 用嵌套乘法计算例 4.10 中的 $P_3(2.5)$。

解:

利用式(6),写出:

$$P_3(x) = ((-0.1(x - 4) + 0.5)(x - 3) - 2)(x - 1) + 5$$

式(7)中的值为:

$$\begin{aligned} S_3 &= -0.1, \\ S_2 &= -0.1(2.5 - 4) + 0.5 = 0.65 \\ S_1 &= 0.65(2.5 - 3) - 2 = -2.325 \\ S_0 &= -2.325(2.5 - 1) + 5 = 1.5125 \end{aligned}$$

故,$P_3(2.5) = 1.5125$。

4.4.2 多项式逼近、节点及中心

假设要找出逼近给定函数 $f(x)$的所有多项式 $P_1(x),\cdots,P_N(x)$的系数 a_k,则 $P_k(x)$应基于中心 $x_0,x_1,\cdots,x_k$ 且有节点$x_0,x_1,\cdots,x_{k+1}$。对多项式 $P_1(x)$,系数 a_0 和 a_1 有类似的含义。在这种情况下,有:

$$P_1(x_0) = f(x_0) \quad 和 \quad P_1(x_1) = f(x_1) \tag{8}$$

利用式(1)和式(8)求解 a_0,得:

$$f(x_0) = P_1(x_0) = a_0 + a_1(x_0 - x_0) = a_0 \tag{9}$$

故 $a_0 = f(x_0)$。然后,利用式(1)、式(8)和式(9),有:

$$f(x_1) = P_1(x_1) = a_0 + a_1(x_1 - x_0) = f(x_0) + a_1(x_1 - x_0)$$

由它可解出 a_1,于是有:

$$a_1 = \frac{f(x_1) - f(x_0)}{x_1 - x_0} \tag{10}$$

故 a_1 是过两点$(x_0,f(x_0))$和$(x_1,f(x_1))$的割线的斜率。

$P_1(x)$和 $P_2(x)$的系数 a_0 和 a_1 相同。在节点 x_2 处计算式(2)，可得：

$$f(x_2)=P_2(x_2)=a_0+a_1(x_2-x_0)+a_2(x_2-x_0)(x_2-x_1) \tag{11}$$

式(9)和式(10)中的值 a_0 和 a_1 可用于式(11)的计算，得：

$$\begin{aligned} a_2 &= \frac{f(x_2)-a_0-a_1(x_2-x_0)}{(x_2-x_0)(x_2-x_1)} \\ &= \left(\frac{f(x_2)-f(x_0)}{x_2-x_0}-\frac{f(x_1)-f(x_0)}{x_1-x_0}\right)\Big/(x_2-x_1) \end{aligned}$$

为了计算方便，该值最好写为：

$$a_2=\left(\frac{f(x_2)-f(x_1)}{x_2-x_1}-\frac{f(x_1)-f(x_0)}{x_1-x_0}\right)\Big/(x_2-x_0) \tag{12}$$

将上两个关于 a_2 的公式写为对公共分母$(x_2-x_1)(x_2-x_0)(x_1-x_0)$的商，可以证明二者等价。式(12)中的分子是一次差商的差，首先我们需要引入差商的概念。

定义 4.1(差商) 函数 $f(x)$的差商定义如下：

$$\begin{aligned} f[x_k] &= f(x_k) \\ f[x_{k-1},x_k] &= \frac{f[x_k]-f[x_{k-1}]}{x_k-x_{k-1}} \\ f[x_{k-2},x_{k-1},x_k] &= \frac{f[x_{k-1},x_k]-f[x_{k-2},x_{k-1}]}{x_k-x_{k-2}} \\ f[x_{k-3},x_{k-2},x_{k-1},x_k] &= \frac{f[x_{k-2},x_{k-1},x_k]-f[x_{k-3},x_{k-2},x_{k-1}]}{x_k-x_{k-3}} \end{aligned} \tag{13}$$

构造高次差商的递归规则为：

$$f[x_{k-j},x_{k-j+1},\cdots,x_k]=\frac{f[x_{k-j+1},\cdots,x_k]-f[x_{k-j},\cdots,x_{k-1}]}{x_k-x_{k-j}} \tag{14}$$

它用来构造表4.8中的差商。

表 4.8 $y=f(x)$的差商表

x_k	$f[x_k]$	$f[\ ,\ ]$	$f[\ ,\ ,\ ]$	$f[\ ,\ ,\ ,\ ]$	$f[\ ,\ ,\ ,\ ,\ ]$
x_0	$f[x_0]$				
x_1	$f[x_1]$	$f[x_0,x_1]$			
x_2	$f[x_2]$	$f[x_1,x_2]$	$f[x_0,x_1,x_2]$		
x_3	$f[x_3]$	$f[x_2,x_3]$	$f[x_1,x_2,x_3]$	$f[x_0,x_1,x_2,x_3]$	
x_4	$f[x_4]$	$f[x_3,x_4]$	$f[x_2,x_3,x_4]$	$f[x_1,x_2,x_3,x_4]$	$f[x_0,x_1,x_2,x_3,x_4]$

$P_N(x)$的系数 a_k 依赖于值$f(x_j)$，$j=0,1,\cdots,k$。下面的定理说明 a_k 可用差商计算：

$$a_k=f[x_0,x_1,\cdots,x_k] \tag{15}$$

定理 4.5(牛顿多项式) 设 $x_0,x_1,\cdots,x_N$ 是$[a,b]$内 $N+1$ 个不同的数，存在至多 N 次的惟一多项式 $P_N(x)$，具有性质：

$$f(x_j)=P_N(x_j),\ j=0,1,\cdots,N$$

该多项式的牛顿形式为:

$$P_N(x)=a_0+a_1(x-x_0)+\cdots+a_N(x-x_0)(x-x_1)\cdots(x-x_{N-1}) \tag{16}$$

其中,$a_k=f[x_0,x_1,\ldots,x_k]$,$k=0,1,\ldots,N$。

注:若$\{(x_j,y_j)\}_j^N=0$是横坐标不同的一组点,则$f(x_j)=y_j$的值可用来构造经过这$N+1$个点的惟一的次数小于等于N的多项式。

推论 4.2(牛顿逼近) 设$P_N(x)$是定理4.5中给出的牛顿多项式,并用来逼近函数$f(x)$,即:

$$f(x)=P_N(x)+E_N(x) \tag{17}$$

若$f\in C^{N+1}[a,b]$,则对$x\in[a,b]$,对应地存在(a,b)内的数$c=c(x)$,使得误差项形如:

$$E_N(x)=\frac{(x-x_0)(x-x_1)\cdots(x-x_N)f^{(N+1)}(c)}{(N+1)!} \tag{18}$$

注:误差项$E_N(x)$与第4.3节中等式(16)和拉格朗日插值误差项相同。

从已知的N次多项式$f(x)$开始计算其差商表是很有意思的,对所有的x有$f^{(N+1)}(x)=0$,而计算显示,第$N+1$个差商为0。这是由于差商(14)正比于j阶导数的数值逼近。

例 4.12 假定$f(x)=x^3-4x$,基于点$x_0=1,x_1=2,\cdots,x_5=6$构造差商表,并求基于节点x_0,x_1,x_2和x_3的牛顿多项式$P_3(x)$。

解见表4.9。

表 4.9 差商表,用于构造例4.12中的牛顿多项式

x_k	$f[x_k]$	第一差商	第二差商	第三差商	第四差商	第五差商
$x_0=1$	<u>-3</u>					
$x_1=2$	0	<u>3</u>				
$x_2=3$	15	15	<u>6</u>			
$x_3=4$	48	33	9	<u>1</u>		
$x_4=5$	105	57	12	1	<u>0</u>	
$x_5=6$	192	87	15	1	0	<u>0</u>

$P_3(x)$的系数$a_0=-3$,$a_1=3$,$a_2=6$,$a_3=1$出现在差商表的对角线上,中心点$x_0=1,x_1=2$和$x_2=3$为第一列中的元素,由式(3)可写出:

$$P_3(x)=-3+3(x-1)+6(x-1)(x-2)+(x-1)(x-2)(x-3)$$

例 4.13 基于5个点$(k,\cos(k))$,$k=0,1,2,3,4$,构造$f(x)=\cos(x)$的差商表,并用它找出$k=1,2,3,4$的系数a_k和4个牛顿插值多项式$P_k(x)$,$k=1,2,3,4$。

解:

为简单起见,将结果四舍五入到小数点后第7位,在表4.10中列出。在式(16)中使用表中的节点x_0,x_1,x_2,x_3和对角线元素a_0,a_1,a_2,a_3,a_4可写出前4个牛顿多项式:

$$P_1(x)=1.0000000-0.4596977(x-0.0)$$

$$P_2(x)=1.0000000-0.4596977(x-0.0)-0.2483757(x-0.0)(x-1.0)$$

$$P_3(x)=1.0000000-0.4596977(x-0.0)-0.2483757(x-0.0)(x-1.0)+0.1465592(x-0.0)(x-1.0)(x-2.0)$$

$$
\begin{aligned}
P_4(x) &= 1.0000000 - 0.4596977(x-0.0) - 0.2483757(x-0.0)(x-1.0) \\
&\quad + 0.1465592(x-0.0)(x-1.0)(x-2.0) \\
&\quad - 0.0146568(x-0.0)(x-1.0)(x-2.0)(x-3.0)
\end{aligned}
$$

表 4.10　差商表,用于构造例 4.13 中的牛顿多项式

x_k	$f[x_k]$	$f[\ ,\ ]$	$f[\ ,\ ,\ ]$	$f[,,,]$	$f[,,,,]$
$x_0=0.0$	1.0000000				
$x_1=1.0$	0.5403023	-0.4596977			
$x_2=2.0$	-0.4161468	-0.9564491	-0.2483757		
$x_3=3.0$	-0.9899925	-0.5738457	0.1913017	0.1465592	
$x_4=4.0$	-0.6536436	0.3363499	0.4550973	0.0879318	-0.0146568

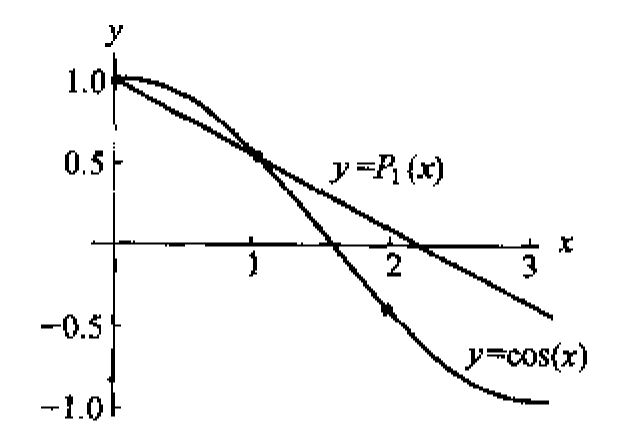
下面的计算实例说明了怎样计算系数 a_2:

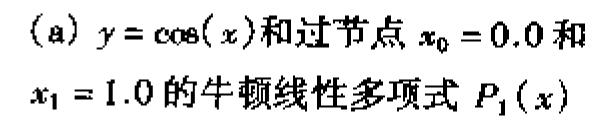
$$f[x_0,x_1]=\frac{f[x_1]-f[x_0]}{x_1-x_0}=\frac{0.5403023-10000000}{1.0-0.0}=-0.4596977$$

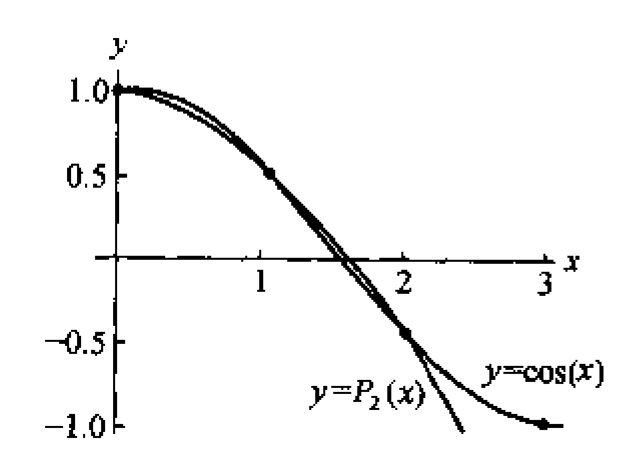
$$f[x_1,x_2]=\frac{f[x_2]-f[x_1]}{x_2-x_1}=\frac{-0.4161468-0.5403023}{2.0-1.0}=-0.9564491$$

$$a_2=f[x_0,x_1,x_2]=\frac{f[x_1,x_2]-f[x_0,x_1]}{x_2-x_0}=\frac{-0.9564491+0.4596977}{2.0-0.0}=-0.2483757$$

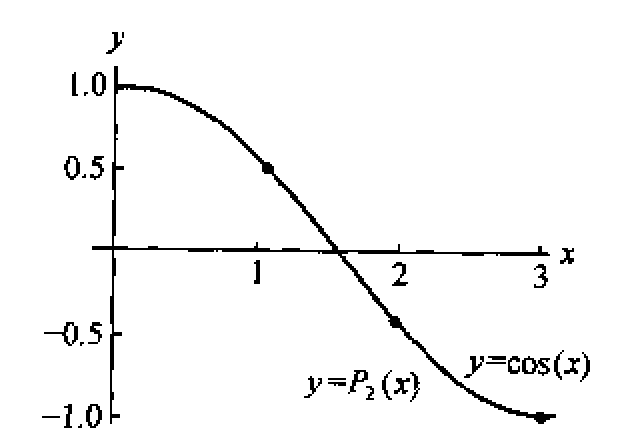
图 4.14(a),(b)和(c)分别给出了 $y=\cos(x)$ 和 $y=P_1(x)$, $y=P_2(x)$, $y=P_3(x)$ 的曲线图。




(a) $y=\cos(x)$ 和过节点 $x_0=0.0$ 和 $x_1=1.0$ 的牛顿线性多项式 $P_1(x)$



(b) $y=\cos(x)$ 和过节点 $x_0=0.0, x_1=1.0$ 和 $x_2=2.0$ 的牛顿 2 次多项式 $y=P_2(x)$



(c) $y=\cos(x)$ 和过节点 $y=P_2(x)$, $x_0=0.0$, $x_1=1.0$ 和 $x_2=2.0$ 的牛顿 3 次多项式 $x_3=3.0$

图 4.14　$y=\cos(x)$ 和不同牛顿多项式的曲线图

为便于计算,表 4.8 中的差商需要存储在数组 $D(k,j)$中:

$$D(k,j) = f[x_{k-j}, x_{k-j+1}, \cdots, x_k] \quad j \leqslant k \tag{19}$$

利用式(14),得到递归计算数组元素的公式:

$$D(k,j) = \frac{D(k,j-1) - D(k-1,j-1)}{x_k - x_{k-j}} \tag{20}$$

注意式(15)中的值 a_k 为对角线元素 $a_k = D(k,k)$。下面给出计算差商及求 $P_N(x)$值的算法,"算法与程序"一节中的问题 2 讨论了如何修改该算法,使得可以用一个一维数组计算值 $\{a_k\}$。

程序 4.2(牛顿插值多项式) 构造和计算经过点$(x_k, y_k) = (x_k, f(x_k))$, $k = 0,1,\cdots,N$ 的次数小于等于 N 的牛顿多项式:

$$P(x) = d_{0,0} + d_{1,1}(x - x_0) + d_{2,2}(x - x_0)(x - x_1) + \cdots + d_{N,N}(x - x_0)(x - x_1)\cdots(x - x_{N-1}) \tag{21}$$

其中:

$$d_{k,0} = y_k, \; d_{k,j} = \frac{d_{k,j-1} - d_{k-1,j-1}}{x_k - x_{k-j}}$$

```
function [C,D] = newpoly(X,Y)
% Input   - X is a vector that contains a list of abscissas
%         - Y is a vector that contains a list of ordinates
% Output  - C is a vector that contains the coefficients
%           of the Newton intepolatory ploynomial
%         - D is the divided-difference table
n = length(X);
D = zeros(n,n);
D(:,1) = Y'
% Use formula (20) to form the divided-difference table
for j = 2:n
    for k = j:n
      D(k,j) = (D(K,j - 1) - D(k - 1,j - 1))/(X(k) - X(k - j + 1));
   end
end
% Determine the coefficients of the Newton interpolating
% polynomial
C = D(n,n);
for k = (n - 1): - 1:1
   C = conv(C,poly(X(k)));
   m = length(C);
   C(m) = C(m) + D(k,k);
end
```

4.4.3 习题

在习题 1 ~ 4 中,利用中心点 x_0, x_1, x_2 及 x_3 系数 a_0, a_1, a_2, a_3 和 a_4 求牛顿多项式 $P_1(x), P_2(x), P_3(x)$ 和 $P_4(x)$,并在点 $x = c$ 处求其值。提示:使用式(1) ~ 式(4)和例

4.9 中的技术。

1. $a_0=4$ $a_1=-1$ $a_2=0.4$ $a_3=0.01$ $a_4=-0.002$
   $x_0=1$ $x_1=3$ $x_2=4$ $x_3=4.5$ $c=2.5$
2. $a_0=5$ $a_1=-2$ $a_2=0.5$ $a_3=-0.1$ $a_4=0.003$
   $x_0=0$ $x_1=1$ $x_2=2$ $x_3=3$ $c=2.5$
3. $a_0=7$ $a_1=3$ $a_2=0.1$ $a_3=0.05$ $a_4=-0.04$
   $x_0=-1$ $x_1=0$ $x_2=1$ $x_3=4$ $c=3$
4. $a_0=-2$ $a_1=4$ $a_2=-0.04$ $a_3=0.06$ $a_4=0.005$
   $x_0=-3$ $x_1=-1$ $x_2=1$ $x_3=4$ $c=2$

在习题 5 ~ 8 中:

(a) 计算函数的差商表。

(b) 写出牛顿多项式 $P_1(x),P_2(x),P_3(x)$和 $P_4(x)$。

(c) 在给定值 x 处求(b) 中牛顿多项式的值。

(d) 比较(c) 中的结果与实际函数值。

5. $f(x)=x^{1/2}$
   $x=4.5,\ 7.5$

k	x_k	$f(x_k)$
0	4.0	2.0000
1	5.0	2.23607
2	6.0	2.44949
3	7.0	2.64575
4	8.0	2.82843

6. $f(x)=3.6/x$
   $x=2.5,\ 3.5$

k	x_k	$f(x_k)$
0	1.0	3.60
1	2.0	1.80
2	3.0	1.20
3	4.0	0.90
4	5.0	0.72

7. $f(x)=3\sin^2(\pi x/6)$
   $x=1.5,\ 3.5$

k	x_k	$f(x_k)$
0	0.0	0.00
1	1.0	0.75
2	2.0	2.25
3	3.0	3.00
4	4.0	2.25

8. $f(x)=e^{-x}$
   $x=0.5,\ 1.5$

k	x_k	$f(x_k)$
0	0.0	1.00000
1	1.0	0.36788
2	2.0	0.13534
3	3.0	0.04979
4	4.0	0.01832

9. 考虑 $M+1$ 个点$(x_0,y_0),\cdots,(x_M,y_M)$。

   (a) 若$(N+1)$阶差商为 0,则证明 $N+2$ 直到 M 阶差商都为 0。

   (b) 若$(N+1)$阶差商为 0,则证明存在一 N 次多项式$P_N(x)$,使得:

$$P_N(x_k)=y_k,\ k=0,1,\cdots,M$$

在习题10~12中,用第9题的结果找出过 $M+1$ 个点($N<M$)的多项式 $P_N(x)$。

10.

x_k	y_k
0	-2
1	2
2	4
3	4
4	2
5	-2

11.

x_k	y_k
1	8
2	17
3	24
4	29
5	32
6	33

12.

x_k	y_k
0	5
1	5
2	3
3	5
4	17
5	45
6	95

13. 利用推论4.2,找出在中心点 $x_0=0, x_1=\pi/2$ 和 $x_2=\pi$ 处,逼近 $f(x)=\cos(\pi x)$ 的牛顿多项式 $P_2(x)$ 在区间 $[0,\pi]$ 内的最大误差($|E_2(x)|$)的界。

4.4.4 算法与程序

1. 用程序4.2重新计算第4.3节的"算法与程序"中的问题2。
2. 在程序4.2中,矩阵 D 用来保存差商表。

   (a) 证明下面的修改是计算牛顿插值多项式的等价方法:

```
for k = 0:N
   A(k) = Y(k);
end
for j = 1:N
    for k = N:-1:j
      A(k) = (A(k) - A(k-1))/(X(k) - X(k-j));
   end
end
```

   (b)利用修改后的程序4.2重新计算问题1。

4.5 切比雪夫多项式(可选)

考虑 $[-1,1]$ 内 $f(x)$ 的基于节点 $-1\leqslant x_0<x_1<\cdots<x_N\leqslant 1$ 的多项式插值,拉格朗日多项式和牛顿多项式都满足:

$$f(x) = P_N(x) + E_N(x)$$

其中:

$$E_N(x) = Q(x)\frac{f^{(N+1)}(c)}{(N+1)!} \tag{1}$$

而 $Q(x)$ 为 $N+1$ 次多项式:

$$Q(x) = (x-x_0)(x-x_1)\cdots(x-x_N) \tag{2}$$

利用关系式:

$$|E_N(x)| \leqslant |Q(x)|\frac{\max_{-1\leqslant x\leqslant 1}\{|f^{(N+1)}(x)|\}}{(N+1)!}$$

我们的任务是根据切比雪夫的推导,选择节点集$\{x_k\}_{k=0}^{N}$,使$\max_{-1\leq x\leq 1}\{|Q(x)|\}$最小。这将需要讨论切比雪夫多项式及其性质,表4.11列出了切比雪夫多项式的前8项。

表4.11 切比雪夫多项式 $T_0(x)$ 到 $T_7(x)$

$T_0(x)=1$
$T_1(x)=x$
$T_2(x)=2x^2-1$
$T_3(x)=4x^3-3x$
$T_4(x)=8x^4-8x^2+1$
$T_5(x)=16x^5-20x^3+5x$
$T_6(x)=32x^6-48x^4+18x^2-1$
$T_7(x)=64x^7-112x^5+56x^3-7x$

4.5.1 切比雪夫多项式性质

性质1 递归关系

切比雪夫多项式可以按如下方法生成。设$T_0(x)=1$和$T_1(x)=x$,利用递归关系:

$$T_k(x)=2xT_{k-1}(x)-T_{k-2}(x),\quad k=2,3,\cdots \tag{3}$$

性质2 导出系数

当$N\geqslant 1$时,$T_N(x)$中x^N的系数为2^{N-1}。

性质3 对称性

当$N=2M$时,$T_{2M}(x)$为偶函数,即:

$$T_{2M}(-x)=T_{2M}(x) \tag{4}$$

当$N=2M+1$时,$T_{2M+1}(x)$为奇函数,即:

$$T_{2M+1}(-x)=-T_{2M+1}(x) \tag{5}$$

性质4 [-1,1]内的三角函数表示

$$T_N(x)=\cos(N\arccos(x)),\ -1\leqslant x\leqslant 1 \tag{6}$$

性质5 [-1,1]内的不同零点

在区间[-1,1]内$T_N(x)$有N个不同的零点x_k(见图4.15):

$$x_k=\cos\left(\frac{(2k+1)\pi}{2N}\right),\ k=0,1,\cdots,N-1 \tag{7}$$

这些值称为切比雪夫点(节点)。

性质6 极值

$$|T_N(x)|\leqslant 1,\ -1\leqslant x\leqslant 1 \tag{8}$$

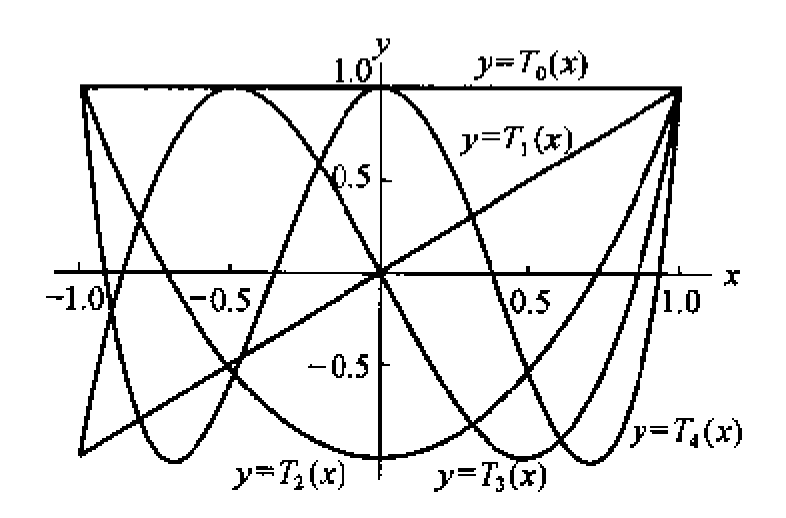


图4.15 [-1,1]内的切比雪夫多项式 $T_0(x), T_1(x), \cdots, T_4(x)$的曲线

性质1通常用作高次切比雪夫多项式的定义,下面我们证明 $T_3(x)=2xT_2(x)-T_1(x)$。利用表4.11中 $T_1(x)$和 $T_2(x)$的表达式,有:

$$2xT_2(x) - T_1(x) = 2x(2x^2-1) - x = 4x^3 - 3x = T_3(x)$$

性质2可通过证明递归关系将 $T_{N-1}(x)$的最高次项系数乘2得到 $T_N(x)$的最高次项系数完成。

性质3通过证明 $T_{2M}(x)$只包含 x 的偶次幂而 $T_{2M+1}(x)$只包含 x 的奇次幂完成。详细证明留作练习。

性质4的证明利用三角恒等式:

$$\cos(k\theta) = \cos(2\theta)\cos((k-2)\theta) - \sin(2\theta)\sin(k-2)\theta)$$

用 $\cos(2\theta)=2\cos^2(\theta)-1$ 和 $\sin(2\theta)=2\sin(\theta)\cos(\theta)$代换,得:

$$\cos(k\theta) = 2\cos(\theta)(\cos(\theta)\cos((k-2)\theta) - \sin(\theta)\sin((k-2)\theta)) - \cos((k-2)\theta)$$

简化得:

$$\cos(k\theta) = 2\cos(\theta)\cos(k-1)\theta) - \cos((k-2)\theta)$$

最后,代入 $\theta=\arccos(x)$得:

$$\begin{aligned}&2x\cos((k-1)\arccos(x)) - \cos((k-2)\arccos(x))\\&= \cos(k\arccos(x)),\ -1 \leqslant x \leqslant 1\end{aligned} \tag{9}$$

最前2个切比雪夫多项式是 $T_0(x)=\cos(0\arccos(x))=1$ 和 $T_1(x)=\cos(1\arccos(x))$,设对 $k=2,3,\cdots,N-1$,有 $T_k(x)=\cos(k\arccos(x)), k=2,3,\cdots,N-1$。将(9)式代入公式(3),得到一般情况:

$$\begin{aligned}T_N(x) &= 2xT_{N-1}(x) - T_{N-2}(x)\\&= 2x\cos((N-1)\arccos(x)) - \cos((N-2)\arccos(x))\\&= \cos(N\arccos(x)),\ -1 \leqslant x \leqslant 1\end{aligned}$$

性质5和性质6是性质4的推论。

4.5.2 最小上界

俄罗斯数学家切比雪夫研究了如何使 $|E_N(x)|$的上界最小。一种方法是使用[-1,1]内 $|Q(x)|$的最大值与[-1,1]内 $|f^{(N+1)}(x)/(N+1)!|$最大值之积。切比雪夫发现,要使因子 $\max\{|Q(x)|\}$最小,应该选择 $x_0, x_1, \cdots, x_N$,使得 $Q(x)=(1/2^N)T_{N+1}(x)$。

定理 4.6 设 N 是固定的,对等式(2)的所有可选的 $Q(x)$,即对 $[-1,1]$ 中所有可能的不同节点 $\{x_k\}_{k=0}^{N}$,多项式 $T(x)=T_{N+1}(x)/2^N$ 是惟一的,具有性质:

$$\max_{-1\leqslant x\leqslant 1}\{|T(x)|\}\leqslant\max_{-1\leqslant x\leqslant 1}\{|Q(x)|\}$$

的选择。并且:

$$\max_{-1\leqslant x\leqslant 1}\{|T(x)|\}=\frac{1}{2^N}\tag{10}$$

证明可在参考文献[29]中找到。

该结果可以叙述为,对$[-1,1]$内的拉格朗日插值 $f(x)=P_N(x)+E_N(x)$,误差界的最小值为:

$$(\max\{|Q(x)|\})(\max\{|f^{(N+1)}(x)/(N+1)!|\})$$

当节点$\{x_k\}$为切比雪夫点(节点)时得到。作为示例,我们来构造一个 $T_{N+1}(x)$拉格朗日系数多项式。首先用等距节点,然后再用切比雪夫节点。$N=3$ 次拉格朗日多项式具有形式:

$$P_3(x)=f(x_0)L_{3,0}(x)+f(x_1)L_{3,1}(x)+f(x_2)L_{3,2}(x)+f(x_3)L_{3,3}(x)\tag{11}$$

4.5.3 等距节点

若$f(x)$由$[-1,1]$内至多 3 次的多项式逼近,则将等距节点 $x_0=-1, x_1=-1/3, x_2=1/3$ 和 $x_3=1$ 代入 4.3 节中的式(8),简化后得到表 4.12 中的系数多项式 $L_{3,k}(x)$。

表 4.12 用来构造 $P_3(x)$的拉格朗日系数多项式,基于等距节点 $x_k=-1+2k/3$

$L_{3,0}(x)=-0.06250000+0.06250000x+0.56250000x^2-0.56250000x^3$
$L_{3,1}(x)=0.56250000-1.68750000x-0.56250000x^2+1.68750000x^3$
$L_{3,2}(x)=0.56250000+1.68750000x-0.56250000x^2-1.68750000x^3$
$L_{3,3}(x)=-0.06250000-0.06250000x+0.56250000x^2+0.56250000x^3$

4.5.4 切比雪夫节点

若$f(x)$由$[-1,1]$内至多 3 次的多项式逼近,使用切比雪夫节点 $x_0=\cos(7\pi/8)$, $x_1=\cos(5\pi/8)$, $x_2=\cos(3\pi/8)$和 $x_3=\cos(\pi/8)$,系数多项式的计算是枯燥的(但可以由计算机来完成),其简化后的系数多项式在表 4.13 中给出。

表 4.13 用于构造 $P_3(x)$的系数多项式,基于切比雪夫节点 $x_k=\cos((7-2k)\pi/8)$

$C_0(x)=-0.10355339+0.11208538x+0.70710678x^2-0.76536686x^3$
$C_1(x)=0.60355339-1.57716102x-0.70710678x^2+1.84775906x^3$
$C_2(x)=0.60355339+1.57716102x-0.70710678x^2-1.84775906x^3$
$C_3(x)=-0.10355339-0.11208538x+0.70710678x^2+0.76536686x^3$

例 4.14 比较分别用表 4.12 和 4.13 的系数多项式得到的$f(x)=e^x$ 的 3 次拉格朗日多项式。

解:

用等距节点,可得多项式:

$$P(x) = 0.99519577 + 0.99904923x + 0.54788486x^2 + 0.17615196x^3$$

这是由函数求值:

$$f(x_0) = e^{(-1)} = 0.36787944 \qquad f(x_1) = e^{(-1/3)} = 0.71653131$$

$$f(x_2) = e^{(1/3)} = 1.39561243 \qquad f(x_3) = e^{(1)} = 2.71828183$$

并利用表 4.12 中的系数多项式 $L_{3,k}(x)$构造线性组合得到:

$$P(x) = 0.36787944L_{3,0}(x) + 0.71653131L_{3,1}(x) + 1.39561243L_{3,2}(x) + 2.71828183L_{3,3}(x)$$

类似地,当使用切比雪夫节点时,得:

$$V(x) = 0.99461532 + 0.99893323x + 0.54290072x^2 + 0.17517569x^3$$

注意系数与 $P(x)$中不同,这是使用了不同节点和函数值的结果:

$$f(x_0) = e^{-0.92387953} = 0.39697597$$

$$f(x_1) = e^{-0.38268343} = 0.68202877$$

$$f(x_2) = e^{0.38268343} = 1.46621380$$

$$f(x_3) = e^{0.92387953} = 2.51904417$$

表 4.13 中的系数多项式 $C_k(x)$用来构造线性组合:

$$V(x) = 0.39697597C_0(x) + 0.68202877C_1(x) + 1.46621380C_2(x) + 2.51904417C_3(x)$$

为了比较 $P(x)$与 $V(x)$的精度,图 4.16(a) 和(b) 中分别绘出其误差函数的曲线。最大误差$|e^x - P(x)|$在 $x = 0.75490129$ 处出现,且:

$$|e^x - P(x)| \leq 0.00998481 \quad , \quad -1 \leq x \leq 1$$

而最大误差$|e^x - V(x)|$在 $x = 1$ 处出现,且:

$$|e^x - V(x)| \leq 0.00665687 \quad , \quad -1 \leq x \leq 1$$

注意:$V(x)$的最大误差约为 $P(x)$误差的 2/3,而且误差在区间内的分布也更均匀。

4.5.5 龙格现象

我们现在更进一步来看使用切比雪夫插值节点的优越性,考虑$[-1,1]$区间内基于等距节点的 $f(x)$的拉格朗日插值。误差 $E_N(x) = f(x) - P_N(x)$是否随着 N 增加而趋近于 0 呢?对于像 $\sin(x)$或 e^x 这样的函数,其所有导数有同样的常数界 M,答案是肯定的。而在一般情况下,答案是否定的,而且很容易找到序列$|P_N(x)|$不收敛的函数。若 $f(x) = 1/(1+12x^2)$,则误差项 $E_N(x)$的最大值当 $N \to \infty$ 时增加。这种不收敛性称为龙格现象(见参考文献[90])。基于 11 个等距节点的 10 次拉格朗日多项式在图 4.17(a)中给出,在区间的端点附近发生了大的振荡,若节点数增加,则振荡变得更剧烈。该问题的出现是由于节点是等距的。

若使用切比雪夫节点来构造 $f(x) = 1/(1+12x^2)$的 10 次插值多项式,误差会较小,如图 4.17(b)所示。在使用切比雪夫节点的条件下,误差 $E_N(x)$将随着 $N \to \infty$ 而趋于 0。一般情况下,若 $f(x)$和 $f'(x)$在$[-1,1]$内连续,则可以证明,切比雪夫插值将产生一个多项式序列$\{P_N(x)\}$,在$[-1,1]$内一致收敛于 $f(x)$。

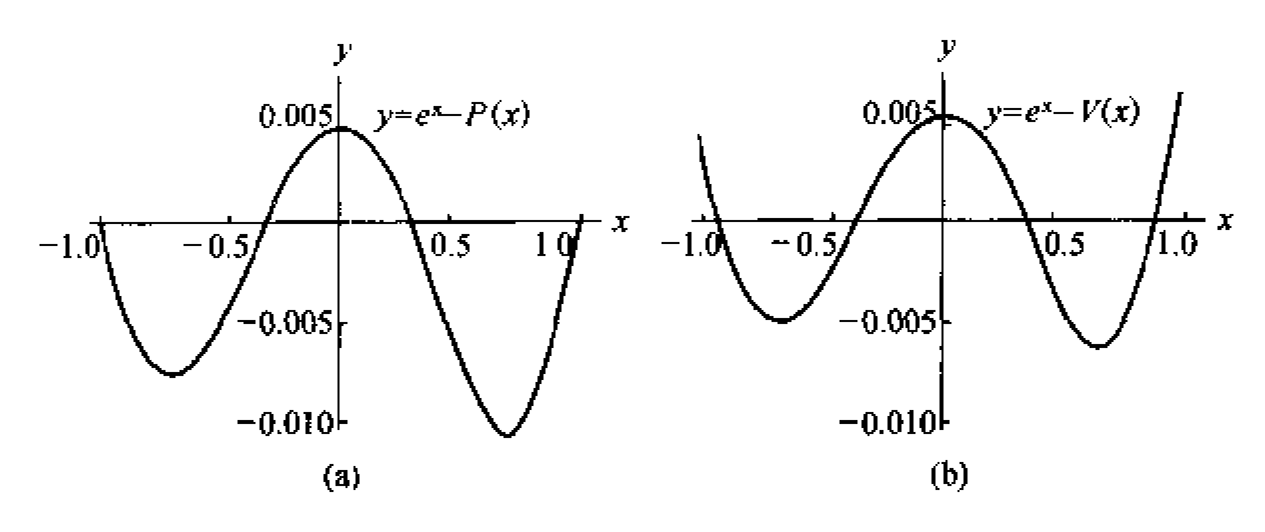
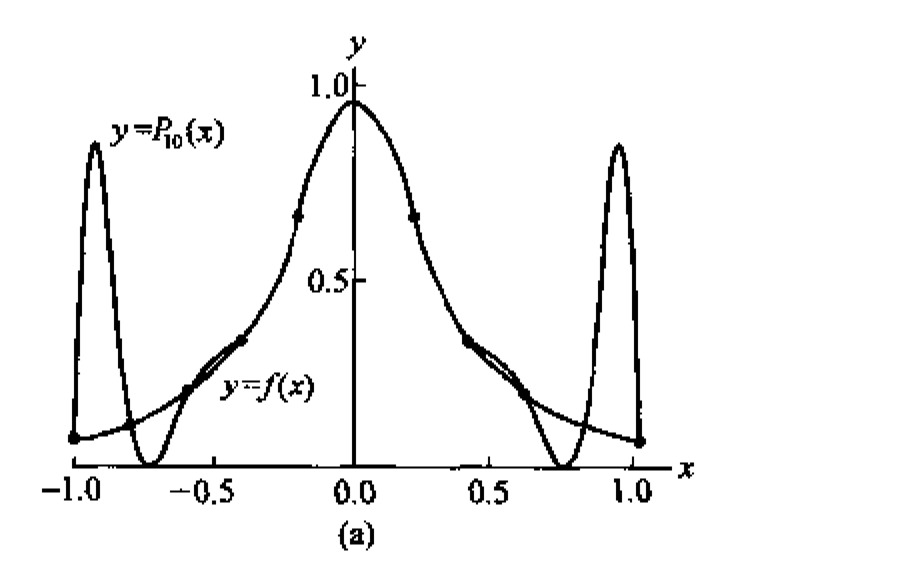


图 4.16　(a) $[-1,1]$内拉格朗日逼近的误差函数 $y=e^x-P(x)$

(b) $[-1,1]$内拉格朗日逼近的误差函数 $y=e^x-V(x)$



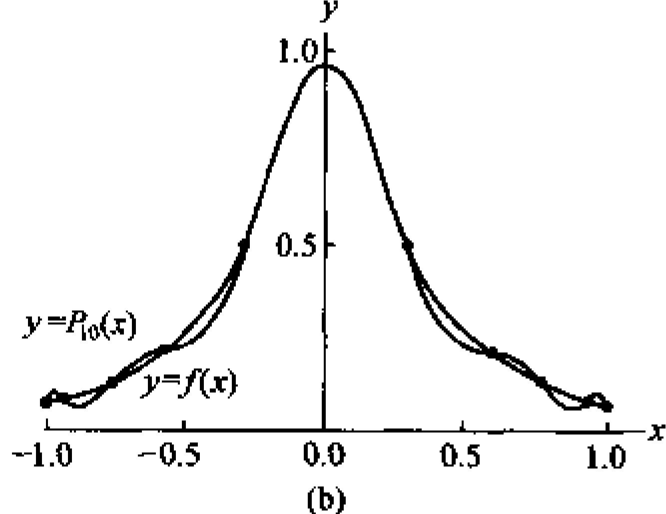


图 4.17　(a) $y=1/(1+12x^2)$的多项式逼近，基于$[-1,1]$内的等距节点

(b) $y=1/(1+12x^2)$的多项式逼近，基于$[-1,1]$内的切比雪夫节点

4.5.6　区间变换

有时需要考察在区间$[a,b]$内描述的问题，并将问题重新在解已知的区间$[c,d]$内表示出来。若要得到区间$[a,b]$内$f(x)$的逼近 $P_N(x)$，则作变量变换，使得问题在区间$[-1,1]$内表示：

$$x=\left(\frac{b-a}{2}\right)t+\frac{a+b}{2}\text{ 或 }t=2\frac{x-a}{b-a}-1 \tag{12}$$

其中 $a\leqslant x\leqslant b$，且 $-1\leqslant t\leqslant 1$。

区间$[-1,1]$内所需的切比雪夫节点为：

$$t_k=\cos\left((2N+1-2k)\frac{\pi}{2N+2}\right),\quad k=0,1,\cdots,N \tag{13}$$

利用式(12)可得$[a,b]$内的插值节点：

$$x_k=t_k\frac{b-a}{2}+\frac{a+b}{2},k=0,1,\cdots,N \tag{14}$$

定理 4.7(拉格朗日-切比雪夫逼近多项式)　设 $P_N(x)$为基于式(14)所给切比雪夫节点的拉格朗日多项式。若 $f\in C^{N+1}[a,b]$，则：

$$|f(x)-P_N(x)|\leqslant\frac{2(b-a)^{N+1}}{4^{N+1}(N+1)!}\max_{a\leqslant x\leqslant b}\{|f^{(N+1)}(x)|\} \tag{15}$$

例 4.15　在$[0,\pi/4]$上对于$f(x)=\sin(x)$，求拉格朗日多项式$P_5(x)$的切比雪夫节点和误差界(15)。

解：

利用式(12)，式(13)和式(14)计算节点：

$$x_k = \cos\left(\frac{(11-2k)\pi}{12}\right)\frac{\pi}{8}+\frac{\pi}{8},\ k=0,1,\cdots,5$$

利用式(15)中的界$|f^{(6)}(x)|\leqslant|-\sin(\pi/4)|=2^{-1/2}=M$，得：

$$|f(x)-P_N(x)|\leqslant\left(\frac{\pi}{8}\right)^6\left(\frac{2}{6!}\right)2^{-1/2}\leqslant 0.00000720$$

4.5.7　正交性质

在例4.14中，利用切比雪夫节点来求拉格朗日插值多项式；通常，这隐含说明N次切比雪夫多项式可由基于$N+1$个节点的拉格朗日多项式求得，这些节点为$T_{N+1}(x)$的$N+1$个根。然而，直接求逼近多项式的方法是将$P_N(x)$展开为表4.11中多项式$T_k(x)$的线性组合。因此，切比雪夫插值多项式可写为：

$$P_N(x)=\sum_{k=0}^{N}c_kT_k(x)=c_0T_0(x)+c_1T_1(x)+\cdots+c_NT_N(x) \tag{16}$$

的形式。

容易求解式(16)中的系数$\{c_k\}$，其技术上的证明需要使用如下的正交性质。设：

$$x_k=\cos\left(\pi\frac{2k+1}{2N+2}\right),\ k=0,1,\cdots,N \tag{17}$$

$$\sum_{k=0}^{N}T_i(x_k)T_j(x_k)=0,\qquad i\neq j \tag{18}$$

$$\sum_{k=0}^{N}T_i(x_k)T_j(x_k)=\frac{N+1}{2},\qquad i=j\neq 0 \tag{19}$$

$$\sum_{k=0}^{N}T_0(x_k)T_0(x_k)=N+1 \tag{20}$$

利用性质4和式(18)及式(20)可以证明如下定理：

定理4.8(切比雪夫逼近)　在$[-1,1]$内$f(x)$的次数小于等于N的切比雪夫逼近多项式$P_N(x)$可写为$\{T_j(x)\}$和的形式：

$$f(x)\approx P_N(x)=\sum_{j=1}^{N}c_jT_j(x) \tag{21}$$

系数$\{c_j\}$可用公式：

$$c_0=\frac{1}{N+1}\sum_{k=0}^{N}f(x_k)T_0(x_k)=\frac{1}{N+1}\sum_{k=0}^{N}f(x_k) \tag{22}$$

和：

$$\begin{aligned}c_j&=\frac{2}{N+1}\sum_{k=0}^{N}f(x_k)T_j(x_k)\\&=\frac{2}{N+1}\sum_{k=0}^{N}f(x_k)\cos\left(\frac{j\pi(2k+1)}{2N+2}\right),j=1,2,\cdots,N\end{aligned} \tag{23}$$

计算。

例 4.16　求区间$[-1,1]$内逼近函数 $P_3(x)$的切比雪夫多项式 $f(x)=e^x$。

解：

利用式(22)和式(23)，以及节点 $x_k:\cos(\pi(2k+1)/8)$，$k=0,1,2,3$ 计算系数：

$$c_0=\frac{1}{4}\sum_{k=0}^{3}e^{x_k}T_0(x_k)=\frac{1}{4}\sum_{k=0}^{3}e^{x_k}=1.26606568$$

$$c_1=\frac{1}{2}\sum_{k=0}^{3}e^{x_k}T_1(x_k)=\frac{1}{2}\sum_{k=0}^{3}e^{x_k}x_k=1.13031500$$

$$c_2=\frac{1}{2}\sum_{k=0}^{3}e^{x_k}T_2(x_k)=\frac{1}{2}\sum_{k=0}^{3}e^{x_k}\cos\left(2\pi\frac{2k+1}{8}\right)=0.27145036$$

$$c_3=\frac{1}{2}\sum_{k=0}^{3}e^{x_k}T_3(x_k)=\frac{1}{2}\sum_{k=0}^{3}e^{x_k}\cos\left(3\pi\frac{2k+1}{8}\right)=0.04379392$$

故，$P_3(x)$的切比雪夫多项式 e^x 为：

$$P_3(x)=1.26606568T_0(x)+1.13031500T_1(x)+0.27145036T_2(x)+0.04379392T_3(x) \tag{24}$$

若将(24)式展开为 x 的幂函数，结果为：

$$P_3(x)=0.99461532+0.99893324x+0.54290072x^2+0.17517568x^3$$

与例 4.14 中的多项式 $V(x)$相同。若目的是求切比雪夫多项式，则最好用式(22)和式(23)。

4.5.8 MATLAB

下面的程序使用 eval 命令，而没有用前面程序中用到的 feval 命令。eval 命令将一个 MATLAB 字符串解释为表达式或语句，例如，下面的命令将快速地计算 $k=0,1,\ldots,5$ 时$x=k/10$处的余弦值：

```
>> x=0:.1:.5;
>>eval('cos(x)')
ans =
   1.0000 0.9950 0.9801 0.9553 0.9211 0.8775
```

程序 4.3(切比雪夫逼近)　构造和计算$[-1,1]$内的 N 次切比雪夫逼近多项式，其中：

$$P(x)=\sum_{j=0}^{N}c_jT_j(x)$$

基于节点：

$$x_k=\cos\left(\frac{(2k+1)\pi}{2N+2}\right)$$

```
function [C,X,Y]=cheby(fun,n,a,b)
% Input   - fun is the string function to be approximated
%         - N is the degree of the chebyshev interpolating
%           polynomial
%         - a is the left end point
%         - b is the right end point
```

```
% Output  - C is the coefficient list for the polynomial
%         - X contains the abscissas
%         - Y contains the ordinates
if nargin= =2,a= -1;b=1;end
d=pi/(2*n+2);
C=zeros(1,n+1);

for k=1:n+1
   X(k)cos((2*k-1)*d);
end
X=(b-a)*X/2+(a+b)/2;
x=X;
Y=eval(fun);

for k =1:n+1
   z=(2*k-1)*d;
   for j=1:n+1
      C(j)=C(j)+Y(k)*cos(j-1)*z);
   end
end
C=2*C/(n+1);
C(1)=C(1)/2;
```

4.5.9 习题

1. 利用性质 1
   (a) 由 $T_3(x)$和 $T_2(x)$构造 $T_4(x)$。
   (b) 由 $T_3(x)$和 $T_4(x)$构造 $T_5(x)$。
2. 利用性质 1
   (a) 由 $T_4(x)$和 $T_5(x)$构造 $T_6(x)$。
   (b) 由 $T_5(x)$和 $T_6(x)$构造 $T_7(x)$。
3. 利用数学归纳法证明性质 2。
4. 利用数学归纳法证明性质 3。
5. 计算区间$[-1,1]$内 T_2 的最大值和最小值[①]。
6. 计算区间$[-1,1]$内 T_3 的最大和最小值。
   提示：$T'_3(1/2)=0$ 和 $T'_3(-1/2)=0$。
7. 计算区间$[-1,1]$内 T_4 的最大和最小值。
   提示：$T'_4(0)=0$, $T'_4(2^{-1/2})=0$ 和 $T'_4(-2^{-1/2})=0$。
8. 设在$[-1,1]$内，$f(x)=\sin(x)$
   (a) 利用表 4.13 中的系数多项式求切比雪夫逼近多项式 $P_3(x)$。
   (b) 求误差界$|\sin(x)-P_3(x)|$。

① 此处当是原文有遗漏，可能是求 $T_2(x)$的最大值和最小值。以下第 6 题和第 7 题类似，分别求 $T_3(x)$和 $T_4(x)$的最小值)——译者注。

9. 设在[-1,1]内,$f(x)=\ln(x+2)$

(a) 利用表 4.13 中的系数多项式求切比雪夫逼近多项式 $P_3(x)$。

(b) 求误差界 $|\ln(x+2)-P_3(x)|$。

10. 2 次拉格朗日多项式具有

$$f(x)=f(x_0)L_{2,0}(x)+f(x_1)L_{2,1}(x)+f(x_2)L_{2,2}(x)$$

的形式,若切比雪夫节点采用 $x_0=\cos(5\pi/6)$, $x_1=0$ 和 $x_2=\cos(\pi/6)$,证明:系数多项式为:

$$L_{2,0}(x)=-\frac{x}{\sqrt{3}}+\frac{2x^2}{3}$$

$$L_{2,1}(x)=1-\frac{4x^2}{3}$$

$$L_{2,2}(x)=\frac{x}{\sqrt{3}}+\frac{2x^2}{3}$$

11. 设在[-1,1]内,$f(x)=\cos(x)$。

(a) 利用习题 10 中的系数多项式求拉格朗日-切比雪夫逼近多项式 $P_2(x)$。

(b) 计算误差界 $|\cos(x)-P_2(x)|$。

12. 设在[-1,1]内,$f(x)=e^x$。

(a) 利用习题 10 中的系数多项式求拉格朗日-切比雪夫逼近多项式 $P_2(x)$。

(b) 计算误差界 $|e^x-P_2(x)|$。

习题 13~15 对[-1,1]内函数 $f(x)$ 的泰勒多项式和拉格朗日-切比雪夫逼近多项式进行比较,计算它们的误差界。

13. $f(x)=\sin(x)$, $N=7$;拉格朗日-切比雪夫逼近多项式为:

$$\sin(x)\approx 0.99999998x-0.16666599x^2+0.00832995x^5-0.00019297x^7$$

14. $f(x)=\cos(x)$, $N=6$;拉格朗日-切比雪夫逼近多项式为:

$$\cos(x)\approx 1-0.49999734x^2+0.04164535x^4-0.00134608x^6$$

15. $f(x)=e^x$, $N=7$;拉格朗日-切比雪夫逼近多项式为:

$$e^x\approx 0.99999980+0.99999998x+0.50000634x^2$$
$$+0.16666737x^3+0.04163504x^4+0.00832984x^5$$
$$+0.00143925x^6+0.00020399x^7$$

16. 证明等式(18)。

17. 证明等式(19)。

4.5.10 算法与程序

在问题 1~6 中,当(a) $N=4$,(b) $N=5$,(c) $N=6$,(d) $N=7$ 时,利用程序 4.3 计算[-1,1]内 $f(x)$ 的切比雪夫多项式 $P_N(x)$ 的系数 $\{c_k\}$。在每种情况中,在同一坐标系中画出 $f(x)$ 和 $P_N(x)$ 的曲线。

1. $f(x)=e^x$
2. $f(x)=\sin(x)$
3. $f(x)=\cos(x)$
4. $f(x)=\ln(x+2)$

5. $f(x)=(x+2)^{1/2}$ 6. $f(x)=(x+2)^{(x+2)}$

7. 利用程序 4.3($N=5$),求 $\int_0^1\cos(x^2)dx$ 的逼近。

4.6 帕德逼近

在本节中,我们引进函数的有理逼近的概念,在其定义域的一个部分区间内逼近函数 $f(x)$。

例如,若 $f(x)=\cos(x)$,则只需找到其在区间$[0,\pi/2]$内的逼近公式,然后可以利用三角恒等式计算$[0,\pi/2]$以外的 $\cos(x)$。

$[a,b]$内的有理函数 $f(x)$ 是两个 N 次和 M 次多项式 $P_N(x)$和 $Q_M(x)$的分式。使用记号 $R_{N,M}(x)$来表示这一分式:

$$R_{N,M}(x)=\frac{P_N(x)}{Q_M(x)}\quad,\quad a\leqslant x\leqslant b \tag{1}$$

我们的目标是使最大误差尽可能地小。对给定的计算能力,通常可以构造出一个有理逼近,其在$[a,b]$内的整体误差小于多项式逼近。以下的推导只是一个导论,且仅限于帕德逼近。

帕德方法要求 $f(x)$及其导数在 $x=0$ 处连续。选择点 $x=0$ 有两个原因,第一,这使计算变得简单;第二,可以通过变量变换把计算平移到一个包含 0 的区间。(1)中使用的多项式为:

$$P_N(x)=p_0+p_1x+p_2x^2+\cdots+p_Nx^N \tag{2}$$

和:

$$Q_M(x)=1+q_1x+q_2x^2+\cdots+q_Mx^M \tag{3}$$

多项式(2)和(3)的构造要求 $f(x)$和 $R_{N,M}(x)$在 $x=0$ 处相等,且其 $N+M$ 阶导数在 $x=0$ 处相等。在 $Q_0(x)=1$ 的情况下,该逼近为 $f(x)$的马克劳林展开。对固定的 $N+M$,当 $P_N(x)$和 $Q_M(x)$有相同次数时或当 $P_N(x)$的次数比 $Q_M(x)$次数高 1 次时误差最小。

注意 Q_M 的常数项为 $q_0=1$,这是允许的,因为它不能为 0,而当 $P_N(x)$和 $Q_M(x)$被同一常数除时,$R_{N,M}(x)$不变。因此有理函数 $R_{N,M}(x)$有 $N+M+1$ 个未知系数。设 $f(x)$是解析的,且有马克劳林展开:

$$f(x)=a_0+a_1x+a_2x^2+\cdots+a_kx^k+\cdots \tag{4}$$

和差 $f(x)Q_M(x)-P_N(x)=Z(x)$:

$$\Big(\sum_{j=0}^{\infty}a_jx^j\Big)\Big(\sum_{j=0}^{M}q_jx^j\Big)-\sum_{j=0}^{N}p_jx^j=\sum_{j=N+M+1}^{\infty}c_jx^j \tag{5}$$

(5)式右端式和式的下标为 $j=N+M+1$,是因为 $f(x)$和 $R_{N,M}(x)$在 $x=0$ 处的前 $N+M$ 阶导数相等。

将(5)式左端展开,且令 $k=0,1,\cdots,N+M$ 的 x^j 系数为 0,可得到线性方程组:

$$a_0-p_0=0$$

$$q_1a_0+a_1-p_1=0$$

$$q_2a_0+q_1a_1+a_2-p_2=0$$

$$q_3a_0+q_2a_1+q_1a_2+a_3-p_3=0$$

$$q_M a_{N-M} + q_{M-1} a_{N-M+1} + \cdots + a_N - P_N = 0 \tag{6}$$

和

$$\begin{aligned} q_M a_{N-M+1} + q_{M-1} a_{N-M+2} + \cdots + q_1 a_N \quad + a_{N+1} &= 0 \\ q_M a_{N-M+2} + q_{M-1} a_{N-M+3} + \cdots + q_1 a_{N+1} \quad + a_{N+2} &= 0 \\ \vdots \qquad\qquad\qquad\qquad\qquad &\vdots \\ q_M a_N \quad + q_{M-1} a_{N+1} \quad + \cdots + q_1 a_{N+M-1} + a_{N+M} &= 0 \end{aligned} \tag{7}$$

注意,在每个等式中,乘式的两个因子的下标和相等,且该和从 0 到 $N+M$ 增大。(7)式中的 M 个等式只包含 N 个未知量 $q_1, q_2, \cdots, q_M$,应该首先求解,然后利用式(6)中的等式求出 $p_0, p_1, \cdots, p_N$。

例 4.17　建立帕德逼近:

$$\cos(x) \approx R_{4,4}(x) = \frac{15\,120 - 6900x^2 + 313x^4}{15\,120 + 660x^2 + 13x^4} \tag{8}$$

解:

在$[-5,5]$区间内的 $\cos(x)$和 $R_{4,4}(x)$曲线见图 4.18。

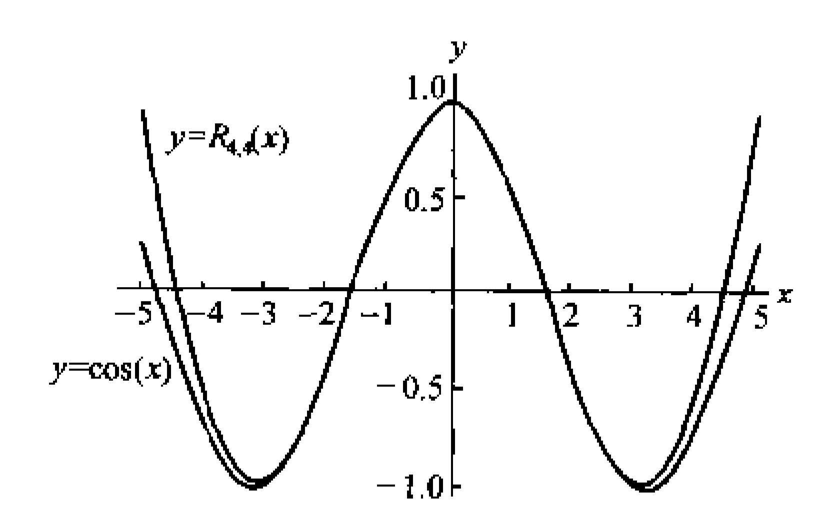


图 4.18　$y=\cos(x)$及其帕德逼近 $R_{4,4}(x)$的曲线图

若使用 $\cos(x)$的马克劳林展开,则将得到包含 9 个未知量的 9 个方程式。而 $\cos(x)$和 $R_{4,4}(x)$都是偶函数且包含 x^2 项,若由 $f(x)=\cos(x^{1/2})$开始则可简化计算:

$$f(x) = 1 - \frac{1}{2}x + \frac{1}{24}x^2 - \frac{1}{720}x^3 + \frac{1}{40\,320}x^4 - \cdots \tag{9}$$

在这种情况下,等式(5)变为:

$$\left(1 - \frac{1}{2}x + \frac{1}{24}x^2 - \frac{1}{720}x^3 + \frac{1}{40\,320}x^4 - \cdots\right)(1 + q_1 x + q_2 x^2) - p_0 - p_1 x - p_2 x^2$$
$$= 0 + 0x + 0x^2 + 0x^3 + 0x^4 + c_5 x^5 + c_6 x^6 + \cdots$$

当比较 x 的前 5 个指数项系数时,得到如下的线性方程组:

$$\begin{aligned} 1 - p_0 &= 0 \\ -\frac{1}{2} + q_1 - p_1 &= 0 \\ \frac{1}{24} - \frac{1}{2}q_1 + q_2 - p_2 &= 0 \end{aligned} \tag{10}$$

$$-\frac{1}{720}+\frac{1}{24}q_1-\frac{1}{2}q_2=0$$

$$\frac{1}{40\,320}-\frac{1}{720}q_1+\frac{1}{24}q_2=0$$

式(10)中的后两个方程必须先求解,它们可改写为易于求解的形式:

$$q_1-12q_2=\frac{1}{30}\quad\text{和}\quad -q_1+30q_2=\frac{-1}{56}$$

先通过等式相加解出 q_2,再求出 q_1:

$$\begin{aligned}q_2&=\frac{1}{18}\left(\frac{1}{30}-\frac{1}{56}\right)=\frac{13}{15\,120}\\ q_1&=\frac{1}{30}+\frac{156}{15\,120}=\frac{11}{252}\end{aligned}\tag{11}$$

利用式(10)的前 3 个方程,显然 $p_0=1$,并且可以利用式(11)中的 q_1 和 q_2 解得 p_1 和 p_2:

$$\begin{aligned}p_1&=-\frac{1}{2}+\frac{11}{252}=-\frac{115}{252}\\ p_2&=\frac{1}{24}-\frac{11}{504}+\frac{13}{15\,120}=\frac{313}{15\,120}\end{aligned}\tag{12}$$

再利用式(11)和式(12)中的系数构造 $f(x)$ 的有理逼近:

$$f(x)\approx\frac{1-115x/252+313x^2/15\,120}{1+11x/252+13x^2/15\,120}\tag{13}$$

由于 $\cos(x)=f(x^2)$,可以用 x^2 代换式(13)中的 x,得到的结果是式(8)中的 $R_{4,4}(x)$ 的公式。

4.6.1 连分式

例 4.17 中的帕德逼近 $R_{4,4}(x)$,每求 1 个值需要至少 12 个算术运算,利用连分式可以将计算量减少到 7 个运算。从(8)式开始,求其多项式余项:

$$\begin{aligned}R_{4,4}(x)&=\frac{15\,120/313-(6900/313)x^2+x^4}{15\,120/13+(660/13)x^2+x^4}\\ &=\frac{313}{13}-\left(\frac{296\,280}{169}\right)\left(\frac{12\,600/823+x^2}{15\,120/13+(600/13)x^2+x^4}\right)\end{aligned}$$

对余项再次进行这一过程,结果为:

$$\begin{aligned}R_{4,4}(x)&=\frac{313}{13}-\frac{296\,280/169}{\dfrac{15\,120/13+(660/13)x^2+x^4}{12\,600/823+x^2}}\\ &=\frac{313}{13}-\frac{296\,280/169}{\dfrac{379\,380}{10\,699}+x^2+\dfrac{420\,078\,960/677\,329}{12\,600/823+x^2}}\end{aligned}$$

为了计算,将分式写为小数形式,得:

$$R_{4,4}(x)=24.07692308-\frac{1753.13609467}{35.45938873+x^2+620.19928277/(15.30984204+x^2)}\tag{14}$$

要计算式(14),首先计算并保存 x^2,然后从分母的最右端开始,依次进行加法,除法,加法,加法,除法,减法。这样总共需要 7 步运算来求得式(14)中的连分式 $R_{4,4}(x)$ 的值。

$R_{4,4}(x)$与 6 次泰勒多项式 $P_6(x)$进行比较,后者也需要 7 步运算来求得嵌套形式:

$$\begin{aligned} P_6(x) &= 1 + x^2\left(-\frac{1}{2} + x^2\left(\frac{1}{24} - \frac{1}{720}x^2\right)\right) \\ &= 1 + x^2(-0.5 + x^2(0.0416666667 - 0.0013888889x^2)) \end{aligned} \tag{15}$$

的值。图 4.19(a) 和(b) 分别给出了$[-1,1]$区间内 $E_R(x) = \cos(x) - R_{4,4}(x)$和 $E_p(x) = \cos(x) - P_6(x)$的曲线图。最大误差在端点处出现,分别为 $E_R(1) = -0.0000003599$ 和 $E_P(1) = 0.0000245281$, $R_{4,4}(x)$的最大误差约为 $P_6(x)$的 1.467%。在较小的区间内,帕德逼近优于泰勒逼近,在$[-0.1,0.1]$区间内,有 $E_R(0.1) = -0.0000000004$ 而 $E_P(0.1) = 0.0000000966$,因此$R_{4,4}(x)$的误差大小约为 $P_6(x)$误差大小的 0.384%。

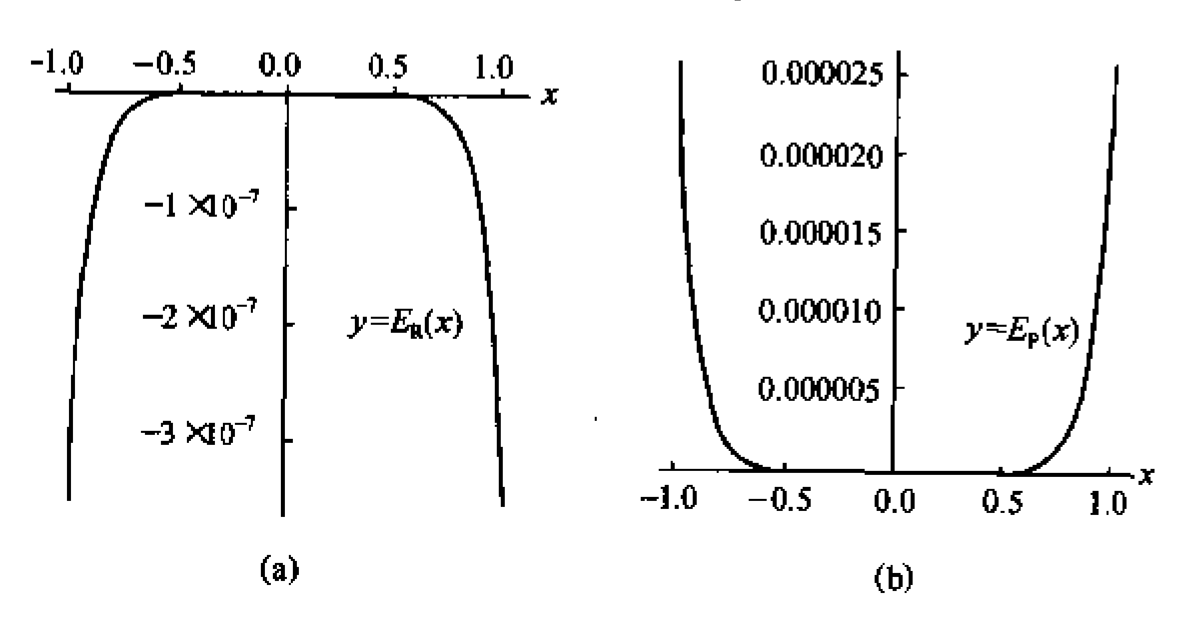


图 4.19　(a) 帕德逼近 $R_{4,4}(x)$的误差 $E_R(x) = \cos(x) - R_{4,4}(x)$曲线
(b) 泰勒逼近 $P_6(x)$的误差 $E_p(x) = \cos(x) = \cos(x) - P_6(x)$曲线

4.6.2　习题

1. 建立帕德逼近:

$$e^x \approx R_{1,1}(x)\frac{2+x}{2-x}$$

2. (a) 求 $f(x) = \ln(1+x)/x$ 的帕德逼近$R_{1,1}(x)$。提示:由马克劳林展开开始:

$$f(x) = 1 - \frac{x}{2} + \frac{x^2}{3} - \cdots$$

(b) 用(a) 中的结果建立逼近:

$$\ln(1+x) \approx R_{2,1}(x) = \frac{6x + x^2}{6 + 4x}$$

3. (a) 求 $f(x) = \tan(x^{1/2})/x^{1/2}$的 $R_{1,1}(x)$。提示:由马克劳林展开开始:

$$f(x) = 1 + \frac{x}{3} + \frac{2x^2}{15} + \cdots$$

(b) 用(a) 中的结果建立逼近:

$$\tan(x) \approx R_{3,2}(x) = \frac{15x - x^3}{15 - 6x^2}$$

4. (a) 求 $f(x) = \arctan(x^{1/2})/x^{1/2}$的 $R_{1,1}(x)$。提示:由马克劳林展开开始:

$$f(x)=1-\frac{x}{3}+\frac{x^2}{5}-\cdots$$

(b) 用(a) 中的结果建立逼近:

$$\arctan(x)\approx R_{3,2}(x)=\frac{15x+4x^3}{15+9x^2}$$

(c) 将(b) 中的有理函数 $R_{3,2}(x)$用连分式形式表示。

5. (a) 建立帕德逼近:

$$e^x\approx R_{2,2}(x)=\frac{12+6x+x^2}{12-6x+x^2}$$

(b) 将(a) 中的有理函数 $R_{2,2}(x)$用连分式形式表示。

6. (a) 求 $f(x)=ln(1+x)/x$ 帕德逼近$R_{2,2}(x)$。提示:由马克劳林展开开始:

$$f(x)=1-\frac{x}{2}+\frac{x^2}{3}-\frac{x^3}{4}+\frac{x^4}{5}-\cdots$$

(b) 利用(a) 中的结果建立:

$$\ln(1+x)\approx R_{3,2}(x)=\frac{30x+21x^2+x^3}{30+36x+9x^2}$$

(c) 将(b) 中的有理函数表示为 $R_{3,2}(x)$连分式形式。

7. (a) 求 $f(x)=\tan(x^{1/2})/x^{1/2}$的 $R_{2,2}(x)$,提示:由马克劳林展开开始:

$$f(x)=1+\frac{x}{3}+\frac{2x^2}{15}+\frac{17x^3}{315}+\frac{62x^4}{2835}+\cdots$$

(b) 利用(a) 中的结果建立:

$$\tan(x)\approx R_{5,4}(x)=\frac{945x-105x^3+x^5}{945-420x^2+15x^4}$$

(c) 将(b) 中的有理函数表示为 $R_{5,4}(x)$连分式形式。

8. (a) 求 $f(x)=\arctan(x^{1/2})/x^{1/2}$的 $R_{2,2}(x)$,提示:由马克劳林展开开始:

$$f(x)=1-\frac{x}{3}+\frac{x^2}{5}-\frac{x^3}{7}+\frac{x^4}{9}-\cdots$$

(b) 利用(a) 中的结果建立:

$$\arctan(x)\approx R_{5,4}(x)=\frac{945x+735x^3+64x^5}{945+1050x^2+225x^4}$$

(c) 将(b) 中的有理函数表示为 $R_{5,4}(x)$连分式形式。

9. 建立帕德逼近:

$$e^x\approx R_{3,3}(x)=\frac{120+60x+12x^2+x^3}{120-60x+12x^2+x^3}$$

10. 建立帕德逼近:

$$e^x\approx R_{4,4}(x)=\frac{1680+840x+180x^2+20x^3+x^4}{1680-840x+180x^2-20x^3+x^4}$$

4.6.3 算法与程序

1. 比较对函数 $f(x)=e^x$ 的逼近:

$$泰勒逼近：T_6(x) = 1 + x + \frac{x^2}{2} + \frac{x^3}{6} + \frac{x^4}{24}$$

$$帕德逼近：R_{2,2}(x) = \frac{12 + 6x + x^2}{12 - 6x + x^2}$$

(a) 在同一坐标系中画出 $f(x)$，$T_6(x)$和 $R_{2,2}(x)$的曲线。

(b) 分别求出在区间[- 1,1]内用 $T_6(x)$和 $R_{2,2}(x)$逼近 $f(x)$的最大误差。

2. 比较对函数 $f(x) = \ln(1 + x)$的逼近：

$$泰勒逼近：T_5(x) = x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \frac{x^5}{5}$$

$$帕德逼近：R_{3,2}(x) = \frac{30x + 21x^2 + x^3}{30 + 36x + 9x^2}$$

(a) 在同一坐标系中画出 $f(x)$，$T_5(x)$和 $R_{3,2}(x)$的曲线。

(b) 分别求出在区间[- 1,1]内用 $T_5(x)$和 $R_{3,2}(x)$逼近 $f(x)$的最大误差。

3. 比较对函数 $f(x) = \tan(x)$的逼近：

$$泰勒逼近：T_9(x) = x + \frac{x^3}{3} + \frac{2x^5}{15} + \frac{17x^7}{315} + \frac{62x^9}{2835}$$

$$帕德逼近：R_{5,4}(x) = \frac{945x - 105x^3 + x^5}{945 - 420x^2 + 15x^4}$$

(a) 在同一坐标系中画出 $f(x)$，$T_9(x)$和 $R_{5,4}(x)$的曲线。

(b) 分别求出在区间[- 1,1]内用 $T_9(x)$和 $R_{5,4}(x)$逼近 $f(x)$的最大误差。

4. 比较在区间[- 1.2,1.2]内对函数 $f(x) = \sin(x)$的帕德逼近：

$$R_{5,4}(x) = \frac{166\,320x - 22\,260x^3 + 551x^5}{15(11\,088 + 364x^2 + 5x^4)}$$

$$R_{7,6}(x) = \frac{11\,511\,339\,840x - 1\,640\,635\,920x^2 + 52\,785\,432x^5 - 479\,249x^7}{7(1\,644\,477\,120 + 39\,702\,960x^2 + 453\,960x^4 + 2\,623x^6)}$$

(a) 在同一坐标系中画出 $f(x)$，$R_{7,6}(x)$和 $R_{5,4}(x)$的曲线。

(b) 分别求出在区间[- 1.2,1.2]内用 $R_{5,4}(x)$和 $R_{7,6}(x)$逼近 $f(x)$的最大误差。

5. (a) 利用式(6)和式(7)导出对函数 $f(x) = \cos(x)$在[- 1.2,1.2]内的逼近 $R_{6,6}(x)$和 $R_{8,8}(x)$。

(b) 在同一坐标系中画出 $f(x)$，$R_{6,6}(x)$和 $R_{8,8}(x)$的曲线。

(c) 分别求出在区间[- 1.2,1.2]内用 $R_{6,6}(x)$和 $R_{8,8}(x)$逼近 $f(x)$的最大误差。

第5章 曲线拟合

在科学技术的工程和试验中，经常需要从试验数据中寻找拟合曲线。例如，在 1601 年，德国天文学家 Johannes Kepler 用公式表示了行星运动第三定律，$T = Cx^{3/2}$，这里 x 表示以百万公里为单位的行星到太阳的距离，T 表示以天为单位的轨道运行周期，而 C 是一个常数。对前 4 个行星，水星、金星、地球和火星的观察数据的(x, T)分别为(55,88)，(108,225)，(150,365)，(228,687)。通过最小二乘法得到的系数 $C = 0.199769$。曲线和数据点如图 5.1 所示。$T = 0.199769x^{3/2}$。

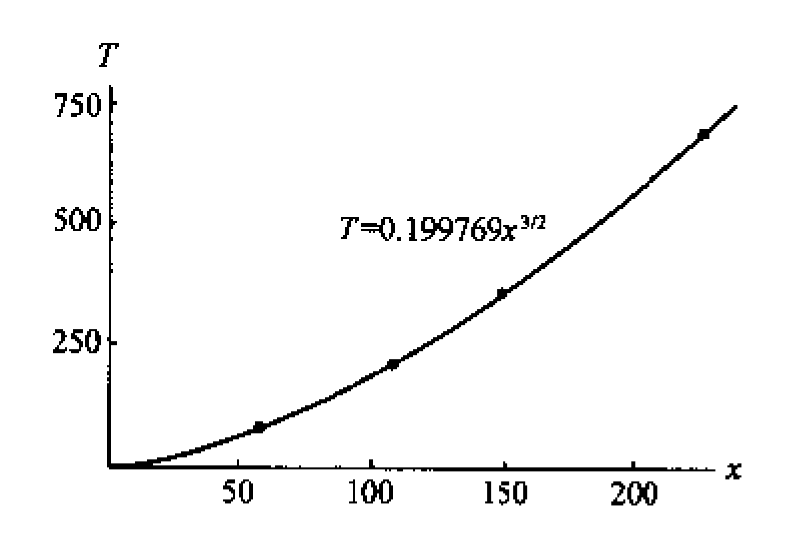


图 5.1 符合开普勒行星运动第三定律的前四个行星的最小二乘拟合曲线 $T = 0.199769x^{3/2}$

5.1 最小二乘拟合曲线

在科学工程试验中，往往会出现这样的情况，实验产生一组数据$(x_1, y_1), \cdots, (x_N, y_N)$，其中横坐标$\{x_k\}$是明确的。而数值方法的目标之一，就是确定一个函数 $y = f(x)$，将这些变量联系起来。通常从一类所允许的函数中进行选择，并最终确定它们的系数。选择函数的可能性是多种多样的。一般会存在一个基于物理情况的底层的数学模型，用于确定函数的形式。在这一节，将主要分析线性函数，其形式为：

$$y = f(x) = Ax + B \tag{1}$$

在第 4 章介绍了如何构造一个经过点集的多项式。如果所有的数值$\{x_k\}$，$\{y_k\}$已知有多位有效数字精度，则能成功地使用多项式插值；否则，不能使用多项式插值。有些试验针对特定的设备设计，因此测试数据点的精度至少有 5 位有效数字。然而，许多试验数据可能只有 3 位或更少的有效数字精度。而且通常在试验中还存在试验误差，所以尽管对$\{x_k\}$，$\{y_k\}$的记录有 3 位有效数字，真实值 $f(x_k)$满足：

$$f(x_k) = y_x + e_k \tag{2}$$

这里 e_k 表示测量误差。

如何找到式(1)的经过测试点附近（不穿过初始点）的最佳线性逼近表达式？要回答这个

问题，首先需要讨论误差（又称偏差或残差）：

$$e_k = f(x_k) - y_x, 1 \leqslant k \leqslant N \tag{3}$$

有多种形式表示式(3)中的误差，可用来测量曲线 $y=f(x)$ 与测试数据的误差。

最大误差：
$$E_\infty(f) = \max_{1\leqslant k\leqslant N}\{|f(x_k) - y_k|\} \tag{4}$$

平均误差：
$$E_1(f) = \frac{1}{N}\sum_{k=1}^{N}|f(x_k) - y_k| \tag{5}$$

均方根误差：
$$E_2(f) = \left(\frac{1}{N}\sum_{k=1}^{N}|f(x_k) - y_k|^2\right)^{1/2} \tag{6}$$

下面的例子显示了当给定一个函数和一组数据后，如何使用这些误差。

表5.1　求例5.1中的 $E_1(f)$ 和 $E_2(f)$ 的计算数据

x_k	y_k	$f(x_k)=8.6-1.6x_k$	$\|e_k\|$	e_k^2
-1	10.0	10.2	0.2	0.04
0	9.0	8.6	0.4	0.16
1	7.0	7.0	0.0	0.00
2	5.0	5.4	0.4	0.16
3	4.0	3.8	0.2	0.04
4	3.0	2.2	0.8	0.64
5	0.0	0.6	0.6	0.35
6	-1.0	-1.0	0.0	0.00
			—	—
			2.6	1.40

例5.1　给定函数 $y=f(x)=8.6-1.6x$ 和一组数据 $(-1,10),(0,9),(1,7),(2,5),(3,4),(4,3),(5,0),(6,-1)$，比较最大误差、平均误差和均方根误差。

根据表5.1中的 $f(x_k)$ 和 e_k 值可求出误差：

$$E_\infty(f) = \max\{0.2,0.4,0.0,0.4,0.2,0.8,0.6,0.0\} = 0.8 \tag{7}$$

$$E_1(f) = \frac{1}{8}(2.6) = 0.325 \tag{8}$$

$$E_2(f) = \left(\frac{1.4}{8}\right)^{1/2} \approx 0.41833 \tag{9}$$

可以看到最大误差值最大，如果有一点的误差严重，则它决定了 $E_\infty(f)$ 的值。平均误差 $E_1(f)$ 简单地将每个点的误差的绝对值进行平均。由于它的简单性，所以常常被使用。误差 $E_2(f)$ 通常用于需要考虑误差的统计特征的情况。

通过求解式(4)到式(6)中某一个的最小值，可得到一条最佳拟合直线。这样可求出三条最佳拟合直线。由于第三个误差方法 $E_2(f)$ 更容易进行最小化计算，所以通常采用它。

5.1.1　求最小二乘曲线

设 $\{(x_k,y_k)\}_{k=1}^{N}$ 是一个具有 N 个点的集合，这里横坐标 $\{x_k\}$ 是确定的。最小二乘拟合曲线 $y=f(x)=Ax+B$ 是满足均方根误差 $E_2(f)$ 最小的曲线。

$E_2(f)$ 的值最小，当且仅当 $N(E_2(f))^2=\sum_{k=1}^{N}(Ax_k+B-y_k)^2$ 的值最小。根据图形的形式，后一个值可解释为：数据点到曲线的垂直距离的平方和的最小值。下面的结论解释了这个过程。

定理 5.1(最小二乘拟合曲线) 设$\{(x_k,y_k)\}_{k=1}^{N}$有 N 个点,其中横坐标$\{x_k\}_{k=1}^{N}$是确定的。最小二乘拟合曲线:

$$y = Ax + B$$

的系数是下列线性方程组的解,称为正规方程:

$$\left(\sum_{k=1}^{N} x_k^2\right)A + \left(\sum_{k=1}^{N} x_k\right)B = \sum_{k=1}^{N} x_y y_k \tag{10}$$
$$\left(\sum_{k=1}^{N} x_k\right)A + NB = \sum_{k=1}^{N} y_k$$

证明:从图形的角度看,对于曲线 $y = Ax + B$,点(x_k, y_k)到线上的点$(x_k, Ax_k + B)$的垂直距离为 $d_k = |Ax_k + B - y_k|$(如图 5.2 所示)。需要对垂直距离的平方和:

$$E(A,B) = \sum_{k=1}^{N}(Ax_k + B - y_k)^2 = \sum_{k=1}^{N} d_k^2 \tag{11}$$

最小化。

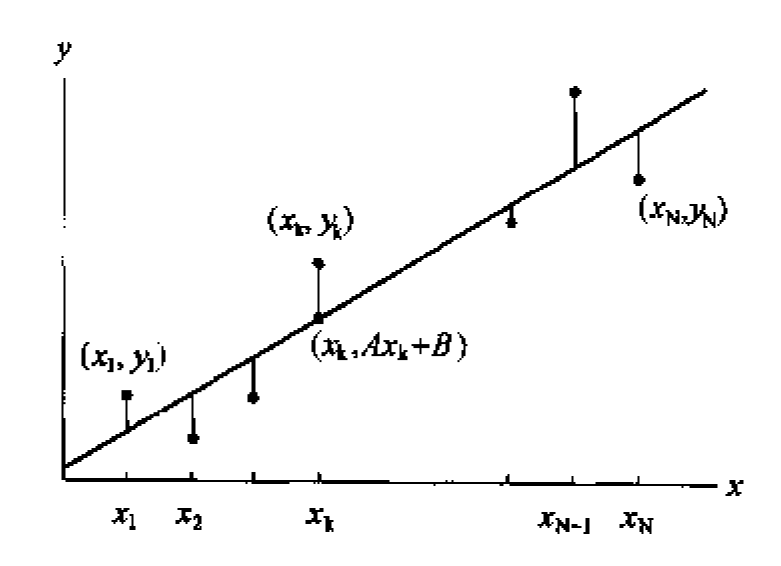


图 5.2 点$\{(x_k,y_k)\}$与最小二乘拟合曲线 $y = Ax + B$ 的垂直距离

通过使偏导数 $\partial E/\partial A$ 和$\partial E/\partial B$ 为零,可得到 $E(A,B)$的最小值,并可求出 A 和B。注意$\{x_k\}$和$\{y_k\}$是式(11)中的常数,而 A 和B 是变量。将 B 固定,对 A 求导可得:

$$\frac{\partial E(A,B)}{\partial A} = \sum_{k=1}^{N} 2(Ax_k + B - y_k)(x_k) = 2\sum_{k=1}^{N}(Ax_k^2 + Bx_k - x_k y_k) \tag{12}$$

现在将 A 固定,对 B 求导可得:

$$\frac{\partial E(A,B)}{\partial B} = \sum_{k=1}^{N} 2(Ax_k + B - y_k) = 2\sum_{k=1}^{N}(Ax_k + B - y_k) \tag{13}$$

令式(12)和式(13)等于零,利用求和的分配律可得:

$$0 = \sum_{k=1}^{N}(Ax_k^2 + Bx_k - x_k y_k) = A\sum_{k=1}^{N} x_k^2 + B\sum_{k=1}^{N} x_k - \sum_{k=1}^{N} x_k y_k \tag{14}$$

$$0 = \sum_{k=1}^{N}(Ax_k + B - y_k) = A\sum_{k=1}^{N} x_k + NB - \sum_{k=1}^{N} y_k \tag{15}$$

可重新排列式(14)和式(15)形成方程组的标准形式,并得到正规方程(10)。通过第 3 章的技术可求出这个线性方程组。然而,程序 5.1 中的方法对数据进行了变换,因此可用良态矩阵(参见练习)。

例5.2 根据例5.1给出的数据点，求其最小二乘拟合曲线。

解：

使用表5.2中的值很容易得到正规方程式(10)中的和。包含 A 和 B 的线性方程组为：

$$92A + 20B = 25$$

$$20A + 8B = 37$$

表5.2 求解正规方程的系数

x_k	y_k	x_k^2	$x_k y_k$
-1	10	1	-10
0	9	0	0
1	7	1	7
2	5	4	10
3	4	9	12
4	3	16	12
5	0	25	0
6	-1	36	-6
20	37	92	25

线性方程组的解为 $A \approx -1.6071429$ 和 $B \approx 8.6428571$。因此最小二乘拟合曲线如图5.3所示，为：

$$y = -1.6071429x + 8.6428571$$

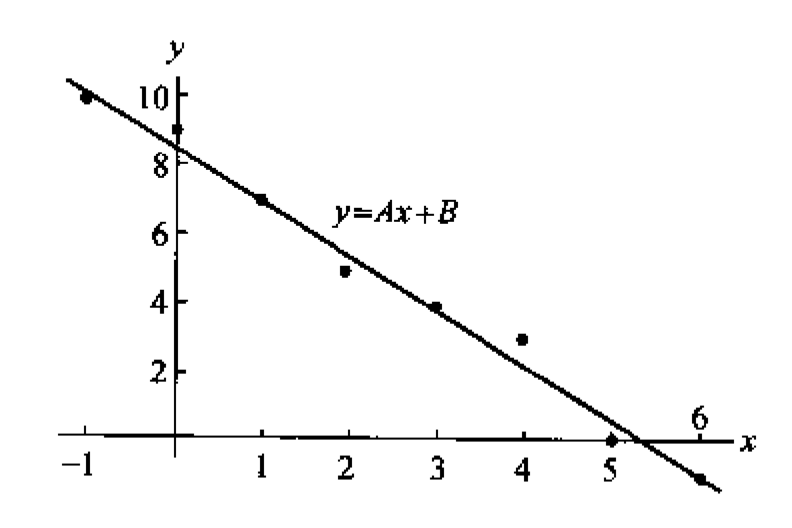


图5.3 最小二乘拟合曲线 $y = -1.6071429x + 8.6428571$

5.1.2 幂函数拟合 $y = Ax^M$

在某些情况下的拟合函数为 $f(x) = Ax^M$，其中 M 是一个已知常数。图5.1中给出的行星运动就是这样的例子。在这种情况下，只需要求解一个参数 A。

定理5.2(幂函数拟合) 设 $\{(x_k, y_k)\}_{k=1}^N$ 为 N 个点，其中横坐标是确定的。最小二乘幂函数拟合曲线 $y = Ax^M$ 的系数 A 为：

$$A = \left(\sum_{k=1}^{N} x_k^M y_k\right) \Big/ \left(\sum_{k=1}^{N} X_k^{2M}\right) \tag{16}$$

使用最小二乘技术，需要求函数 $E(A)$ 的最小值：

$$E(A) = \sum_{k=1}^{N} (Ax_k^M - y_k)^2 \tag{17}$$

在这种情况下，只需求解 $E'=(A)=0$。导数表示为：

$$E'(A) = 2\sum_{k=1}^{N}(Ax_k^M - y_k)(x_k^M) = 2\sum_{k=1}^{N}(Ax_k^{2M} - x_k^M y_k) \tag{18}$$

因此，系数 A 是下面方程的解：

$$0 = A\sum_{k=1}^{N}x_k^{2M} - \sum_{k=1}^{N}x_k^M y_k \tag{19}$$

上式可化简为式(16)。

例 5.3 试验数据如表 5.3 所示。关系式为 $d=\frac{1}{2}gt^2$，d 表示单位为米的距离，t 表示单位为秒的时间。求重力常数 g。

表 5.3 求解幂函数拟合的系数

时间，t_k	距离，d_k	$d_k t_k^2$	t_k^4
0.200	0.1960	0.00784	0.0016
0.400	0.7850	0.12560	0.0236
0.600	1.7665	0.63594	0.1296
0.800	3.1405	2.00992	0.4096
1.000	4.9075	4.90750	1.0000
		7.68680	1.5664

解：

可用表 5.3 中的值求出公式(16)需要的和，这里幂 $M=2$。

系数 $A=7.68680/1.5664=4.9073$，而且可得 $d=4.9073t^2$ 和 $g=2A=9.7146\text{m/s}^2$。

下面的构造最小二乘拟合曲线的程序是计算稳定的：当正规方程式(10)是病态时，可给出可靠解。要求读者用这个程序开发用于练习 4 到练习 7 的算法。

> **程序 5.1(最小二乘拟合曲线)** 根据 N 个数据点 $(x_1,y_1),\cdots,(x_N,y_N)$ 构造最小二乘拟合曲线 $y=Ax+B$

```
function [A,B] = lsline(X,Y)
% Input   - X is the 1xn abscissa vector
%         - Y is the 1xn ordinate vector
% Output  - A is the coefficient of x in Ax + B
%         - B is the constant coefficient in Ax + B
xmean = mean(X);
ymean = mean(Y);
sumx2 = (X - xmean) * (X - xmean)';
sumxy = (Y - ymean) * (X - xmean)';
A = sumxy/sumx2;
B = ymean - A * xmean;
```

5.1.3 最小二乘拟合曲线的练习

在练习 1 和练习 2 中，根据给出的数据点求出最小二乘拟合曲线 $y=f(x)=Ax+B$，并计

算 $E_2(f)$。

1.(a)

x_k	y_k	$f(x_k)$
-2	1	1.2
-1	2	1.9
0	3	2.6
1	3	3.3
2	4	4.0

(b)

x_k	y_k	$f(x_k)$
-6	7	7.0
-2	5	4.6
0	3	3.4
2	2	2.2
6	0	-0.2

(c)

x_k	y_k	$f(x_k)$
-4	-3	-3.0
-1	-1	-0.9
0	0	-0.2
2	1	1.2
3	2	1.9

2.(a)

x_k	y_k	$f(x_k)$
-4	1.2	0.44
-2	2.8	3.34
0	6.2	6.24
2	7.8	9.14
4	13.2	12.04

(b)

x_k	y_k	$f(x_k)$
-6	-5.3	-6.00
-2	-3.5	-2.84
0	-1.7	-1.26
2	0.2	0.32
6	4.0	3.48

(c)

x_k	y_k	$f(x_k)$
-8	6.8	7.32
-2	5.0	3.81
0	2.2	2.64
4	0.5	0.30
6	-1.3	-0.87

3. 根据给出的数据点,求幂函数拟合曲线 $y = Ax$,并计算 $E_2(f)$。其中 $M = 1$,它实际上是经过原点的直线。

(a)

x_k	y_k	$f(x_k)$
-4	-3	-2.8
-1	-1	-0.7
0	0	0.0
2	1	1.4
3	2	2.1

(b)

x_k	y_k	$f(x_k)$
3	1.6	1.722
4	2.4	2.296
5	2.9	2.870
6	3.4	3.444
8	4.6	4.592

(c)

x_k	y_k	$f(x_k)$
1	1.6	1.58
2	2.8	3.16
3	4.7	4.74
4	6.4	6.32
5	8.0	7.90

4. 用下列表达式定义点集 $\{(x_k, y_k)\}_{k=1}^{N}$ 的均值 $\bar{x}$ 和 $\bar{y}$:

$$\bar{x} = \frac{1}{N}\sum_{k=1}^{N} x_k \text{ 和 } \bar{y} = \frac{1}{N}\sum_{k=1}^{N} y_k$$

证明点 $(\bar{x}, \bar{y})$ 位于由点集得到的最小二乘拟合曲线上。

5. 证明式(10)中的方程组的解为:

$$A = \frac{1}{D}\left(N\sum_{k=1}^{N} x_k y_k - \sum_{k=1}^{N} x_k \sum_{k=1}^{N} y_k\right)$$

$$B = \frac{1}{D}\left(\sum_{k=1}^{N} x_k^2 \sum_{k=1}^{N} y_k - \sum_{k=1}^{N} x_k \sum_{k=1}^{N} x_k y_k\right)$$

其中:

$$D = N\sum_{k=1}^{N} x_k^2 - \left(\sum_{k=1}^{N} x_k\right)^2$$

提示:对式(10)中的方程组使用高斯消去法。

6. 证明练习 5 中的 D 非零。

提示：证明 $D = N\sum_{k=1}^{N}(x_k - \bar{x})^2$

7. 证明最小二乘拟合曲线的系数 A 和 B 可用如下方法计算。首先计算练习 4 中的平均值 $\bar{x}$ 和 $\bar{y}$，然后计算：

$$C = \sum_{k=1}^{N}(x_k - \bar{x})^2, A = \frac{1}{C}\sum_{k=1}^{N}(x_k - \bar{x})(y_k - \bar{y}), B = \bar{y} - A\bar{x}$$

提示：设 $X_k = x_k - \bar{x}, Y_k = y_k - \bar{y}$，并求直线 $Y = AX$。

8. 根据下列数据，并使用 $E_2(f)$，求解幂函数拟合曲线 $y = Ax^2$ 和 $y = Bx^3$，并比较哪条曲线更好。

(a)

x_k	y_k
2.0	5.1
2.3	7.5
2.6	10.6
2.9	14.4
3.2	19.0

(b)

x_k	y_k
2.0	5.9
2.3	8.3
2.6	10.7
2.9	13.7
3.2	17.0

9. 根据下列数据，并使用 $E_2(f)$，求解幂函数拟合曲线 $y = A/x$ 和 $y = B/x^2$，并比较哪条曲线更好。

(a)

x_k	y_k
0.5	7.1
0.8	4.4
1.1	3.2
1.8	1.9
4.0	0.9

(b)

x_k	y_k
0.7	8.1
0.9	4.9
1.1	3.3
1.6	1.6
3.0	0.5

10. (a) 推导求解 $y = Ax$ 的最小二乘线性拟合的正规方程。

(b) 推导求解最小二乘幂函数曲线拟合 $y = Ax^2$ 的正规方程。

(c) 推导求解最小二乘抛物线函数曲线拟合 $y = Ax^2 + B$ 的正规方程。

11. 设根据点集 $S_N = \left\{\left(\frac{k}{N}, \left(\frac{k}{N}\right)^2\right)\right\}_{k=1}^{N}$，$N = 2,3,4,\cdots$ 构造最小二乘线性拟合。注意 S_N 中的每个值位于在区间[0,1]内的曲线 $f(x) = x^2$ 上。设 $\bar{x}_N$ 和 $\bar{y}_N$ 是给定数据点的平均值(参见练习 4)。用 $\hat{x}$ 表示在区间[0,1]内的 x 的平均值，用 $\hat{y}$ 表示在区间[0,1]内的 $f(x) = x^2$ 的平均值。

(a) 证明 $\lim_{N\to\infty}\bar{x}_N = \hat{x}$。

(b) 证明 $\lim_{N\to\infty}\bar{y}_N = \hat{y}$。

12. 设根据下列点集构造最小二乘线性拟合：

$$S_N = \left\{\left((b-a)\frac{k}{N} + a, f\left((b-a)\frac{k}{N} + a\right)\right)\right\}_{k=1}^{N}$$

其中 $N = 2,3,4,\cdots$，设 $y = f(x)$ 为在闭区间[a, b]内的可积分函数。重新求解练习 11 的(a)和(b)。

5.1.4 算法和程序

1. 库克定律指出 F = kx，其中 F 是拉伸弹簧的拉力(单位为盎司)，x 是拉伸的长度(单位为英寸)。根据下列试验数据，使用程序 5.1 求解拉伸常量 k 的近似值。

(a)

x_k	F_k
0.2	3.6
0.4	7.3
0.6	10.9
0.8	14.5
1.0	18.2

(b)

x_k	F_k
0.2	5.3
0.4	10.6
0.6	15.9
0.8	21.2
1.0	26.4

2. 根据下列数据，使用例 5.3 中的幂曲线拟合，编写一个程序求解重力常量 g。

(a)

时间，t_k	距离，d_k
0.200	0.1960
0.400	0.7835
0.600	1.7630
0.800	3.1345
1.000	4.8975

(b)

时间，t_k	距离，d_k
0.200	0.1965
0.400	0.7855
0.600	1.7675
0.800	3.1420
1.000	4.9095

3. 下列数据给出了 9 大行星到太阳的距离，以及它们以天为单位的恒星周期。

行星	到太阳的距离(km × 10^6)	恒星周期(天)
水星	57.59	87.99
金星	108.11	224.70
地球	149.57	365.26
火星	227.84	686.98
木星	778.14	4 332.4
土星	1427.0	10 759
天王星	2870.3	30 684
海王星	4499.9	60 188
冥王星	5909.0	90 710

修改问题 2 中的程序，计算 $E_2(f)$，并使用它求解拟合：

(a) 前 4 个行星

(b) 所有 9 个行星

的行星第三运动定律的最小二乘幂函数拟合曲线 $y = Cx^{3/2}$。

4. (a) 根据数据 $\{(x_k, y_k)\}_{k=1}^{50}$，其中 $x_k = (0.1)k$ 且 $y_k = x_k + cos(k^{1/2})$，求解最小二乘线性拟合。

(b) 计算 $E_2(f)$。

(c) 在同一坐标下，画出点集和最小二乘线性拟合曲线。

5.2 曲线拟合

5.2.1 对 $y = Ce^{Ax}$ 线性化方法

设给定点集$(x_1, y_1), (x_2, y_2), \cdots, (x_N, y_N)$,求指数函数的曲线拟合:

$$y = Ce^{Ax} \tag{1}$$

第一步是对式(1)两边取对数:

$$\ln(y) = Ax + \ln(C) \tag{2}$$

然后引入变量变换:

$$Y = \ln(y), X = x, B = \ln(C) \tag{3}$$

变量变换形成线性关系式:

$$Y = AX + B \tag{4}$$

在xy平面上的初始点集$(x_k, y_k) = (x_k, \ln(y_k))$变换成在XY平面上的点集$(X_k, Y_k)$。这个过程称为数据线性化。这样可用最小二乘曲线式(4)拟合点集$\{X_k, Y_k\}$。求解 A 和 B 的正规方程为:

$$\begin{aligned} \left(\sum_{k=1}^{N} X_k^2\right)A + \left(\sum_{k=1}^{N} X_k\right)B &= \sum_{k=1}^{N} X_k Y_k \\ \left(\sum_{k=1}^{N} X_k\right)A + NB &= \sum_{k=1}^{N} Y_k \end{aligned} \tag{5}$$

求出 A 和 B 后,式(1)中的参数 C 可用下式计算:

$$C = e^B \tag{6}$$

例5.4 根据5个点(0,1.5),(1,2.5),(2,3.5),(3,5.0),(4,7.5),使用数据线性化方法求解指数曲线拟合 $y = Ce^{Ax}$。

使用变换公式(3)将初始点变换成:

$$\begin{aligned} \{(X_k, Y_k)\} &= \{0, \ln(1.5), (1, \ln(2.5)), (2, \ln(3.5)), (3, \ln(5.0)), (4, \ln(7.5))\} \\ &= \{(0, 0.40547), (1, 0.91629), (2, 1.25276), (3, 1.60944), (4, 2.01490)\} \end{aligned} \tag{7}$$

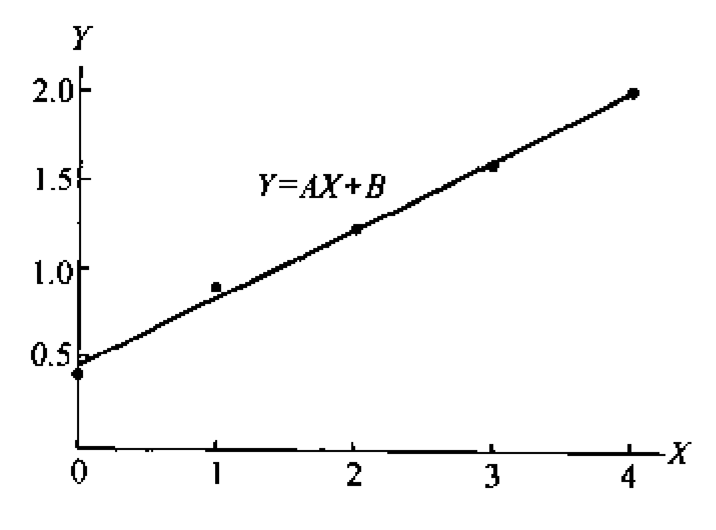


图5.4 变换后的点集$\{(X_k, Y_k)\}$

这些变换后的点如图5.4所示,并具有直线形式。在图5.4中,拟合式(7)中的点的最小二乘曲线 $Y = AX + B$ 为:

$$Y = 0.391202X + 0.457367 \tag{8}$$

计算式(5)中正规方程系数的过程如表5.4所示。

求解 A 和 B 的线性方程组(5)：

$$30A + 10B = 16.309742 \qquad (9)$$
$$10A + 5B = 6.198860$$

表 5.4　根据变换后的数据点集求正规方程的系数

x_k	y_k	X_k	$Y_k = \ln(y_k)$	x_k^2	X_kY_k
0.0	1.5	0.0	0.405 465	0.0	0.000 000
1.0	2.5	1.0	0.916 291	1.0	0.916 291
2.0	3.5	2.0	1.252 763	4.0	2.505 526
3.0	5.0	3.0	1.609 438	9.0	4.828 314
4.0	7.5	4.0	2.014 903	16.0	8.05 9612
		$10.0 = \sum X_k$	$6.198860 = \sum Y_k$	$30.0 = \sum X_k^2$	$16.309743 = \sum X_kY_k$

解为 $A = 0.3912023$ 和 $B = 0.457367$。通过计算可得 $C = e^{0.457367}$，将 A 和 C 代入式(1)，可得到指数曲线拟合(如图 5.5 所示)：

$$y = 1.579910e^{0.3912023x} \qquad \text{(通过数据线性化进行拟合)} \qquad (10)$$

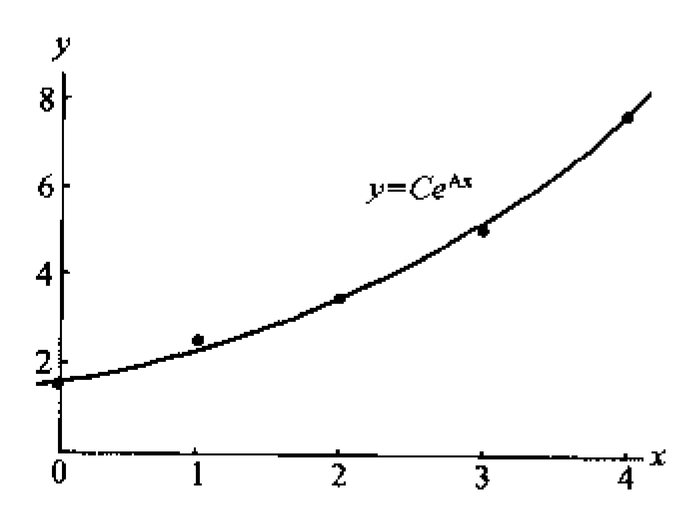


图 5.5　通过数据线性化方法得到的指数曲线拟合 $y = 1.579910e^{0.3912023x}$

5.2.2　求解 $y = Ce^{Ax}$ 的非线性最小二乘法

设有给定点集$(x_1, y_1), (x_2, y_2), \cdots, (x_N, y_N)$，需要拟合指数曲线：

$$y = Ce^{Ax} \qquad (11)$$

采用非线性最小二乘法需要求下式的最小值：

$$E(A, C) = \sum_{k=1}^{N} (Ce^{Ax_k} - y_k)^2 \qquad (12)$$

$E(A, C)$对 A 和 C 的偏导数分别为：

$$\frac{\partial E}{\partial A} = 2\sum_{k=1}^{N} (Ce^{Ax_k} - y_k)(Cx_ke^{Ax_k}) \qquad (13)$$

和：

$$\frac{\partial E}{\partial C} = 2\sum_{k=1}^{N} (Ce^{Ax_k} - y_k)(e^{Ax_k}) \qquad (14)$$

令式(13)和式(14)中偏导数为零，对其进行简化后，可得正规方程：

$$C\sum_{k=1}^{N} x_ke^{2Ax_k} - \sum_{k=1}^{N} x_ky_ke^{Ax_k} = 0$$

$$C\sum_{k=1}^{N} e^{Ax_k} - \sum_{k=1}^{N} y_k e^{Ax_k} = 0 \tag{15}$$

式(15)中的方程对于未知数 A 和 C 是非线性的,可用牛顿法求解。这个计算过程较为费时,而且其中的迭代需要良好的 A 和 C 的初始值。许多软件包有内部的求多变量函数最小值的子程序,可直接用来求解 $E(A,C)$的最小值。例如,Nelder - Mead 单纯形算法可直接用来求解式(12)的最小值,而不用计算式(13)到式(15)。

例 5.5 根据 5 个数据点(0,1.5),(1,2.5),(2,3.5),(3,5.0),(4,7.5),利用最小二乘法求解指数拟合 $y = Ce^{Ax}$。

解:

首先求解 $E(A,C)$的最小值,$E(A,C)$为:

$$E(A,C) = (C-1.5)^2 + (Ce^{A} - 2.5)^2 + (Ce^{2A} - 3.5)^2 + (Ce^{3A} - 5.0)^2 + (Ce^{4A} - 7.5)^2 \tag{16}$$

使用 MATLAB 中的 fmins 命令求解最小化 $E(A,C)$后的 A 和 C 的近似值。首先在 MATLAB 中将 $E(A,C)$定义为一个 M 文件:

```
function z = E(u)
A = u(1);
C = u(2);
z = (c - 1.5).^2 + (C. * exp(A) - 2.5).^2 + (c. * exp(2 * A) - 3.5).^2 + ...
   (C. * exp(3 * A) - 5.0).^2 + (C. * exp(4 * A) - 7.5).^2;
```

在 MATLAB 的命令窗口,使用 fmins 命令和初始值 A = 1.0,C = 1.0,可得:

```
> > fmins('E',[1 1])
ans =
    0.38357046980073 1.61089952247928
```

则 5 个数据点的曲线拟合为:

$$y = 1.610899e^{0.3835705} \quad (\text{非线性最小二乘拟合}) \tag{17}$$

对使用数据线性化和非线性最小二乘法结果的比较如表 5.5 所示。可看到在系数上有一些不同。根据插值法的观点,在区间[0,4]内,误差不超过 2%(如表 5.5 和图 5.6 所示)。如果数据中的误差满足正态分布,则式(17)是更好的选择。在选择数据的范围外进行外推,则两个解将发散,当 $x = 10$ 时,误差增加到大约 6%。

表 5.5 两个指数拟合的比较

x_k	y_k	$1.5799e^{0.39120x}$	$1.6109e^{0.38357x}$
0.0	1.5	1.5799	1.6109
1.0	2.5	2.3363	2.3640
2.0	3.5	3.4548	3.4692
3.0	5.0	5.1088	5.0911
4.0	7.5	7.5548	7.4713
5.0		11.1716	10.9644
6.0		16.5202	16.0904
7.0		24.4293	23.6130
8.0		36.1250	34.6527
9.0		53.4202	50.8535
10.0		78.9955	74.6287

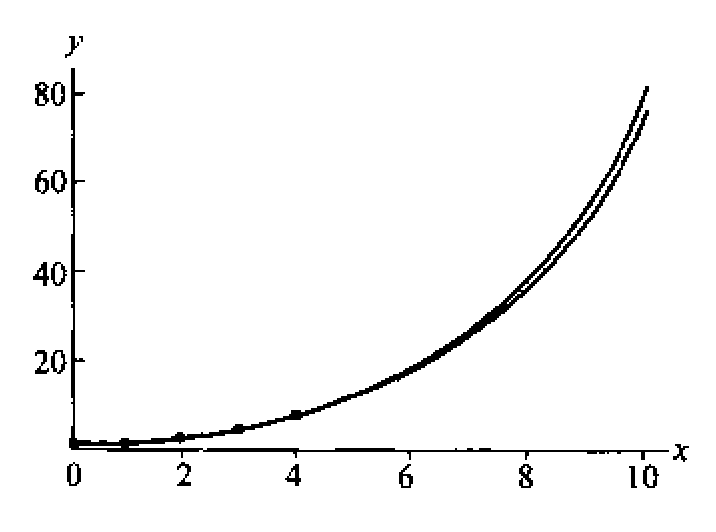


图 5.6　两个指数曲线的图形比较

5.2.3　数据线性化变换

科学家通常使用数据线性化技术来拟合各种曲线，如 $y = Ce^{(Ax)}$，$y = A\ln(x) + B$ 和 $y = A/x + B$。当选定曲线后，必须对变量找到一个合适的变换，以得到线性表达式。例如，读者可验证通过变量(与常量)变换 $X = xy, Y = y, C = -1/A, D = -B/A$。$y = D/(x + C)$可变换成线性表达式 $Y = AX + B$。图 5.7 中显示了多种可能的曲线，其他一些变换如表 5.6 所示。

表 5.6　在数据线性化中的变量替换

函数，$y = f(x)$	线性化性质，$Y = Ax + B$	变量与常数的变化
$y = \frac{A}{x} + B$	$y = A\frac{1}{x} + B$	$X = \frac{1}{x}, Y = y$
$y = \frac{D}{x + C}$	$y = \frac{-1}{C}(xy) + \frac{D}{C}$	$X = xy, Y = y$ $C = \frac{-1}{A}, D = \frac{-B}{A}$
$y = \frac{1}{Ax + B}$	$\frac{1}{y} = Ax + B$	$X = x, Y = \frac{1}{y}$
$y = \frac{x}{Ax + B}$	$\frac{1}{y} = A\frac{1}{x} + B$	$X = \frac{1}{x}, Y = \frac{1}{y}$
$y = A\ln(x) + B$	$y = A\ln(x) + B$	$X = \ln(x), Y = y$
$y = Ce^{Ax}$	$\ln(y) = Ax + \ln(C)$	$X = x, Y = \ln(y)$ $C = e^B$
$y = Cx^A$	$\ln(y) = A\ln(x) + \ln(C)$	$X = \ln(x), Y = \ln(y)$ $C = e^B$
$y = (Ax + B)^{-2}$	$y^{-1/2} = Ax + B$	$X = x, Y = y^{-1/2}$
$y = Cxe^{-Dx}$	$\ln\left(\frac{y}{x}\right) = -Dx + \ln(C)$	$X = x, Y = \ln\left(\frac{y}{x}\right)$ $C = e^B, D = -A$
$y = \frac{L}{1 + Ce^{Ax}}$	$\ln\left(\frac{L}{y} - 1\right) = Ax + \ln(C)$	$X = x, Y = \ln\left(\frac{L}{y} - 1\right)$ $C = e^B$ 且 L是给定的常数

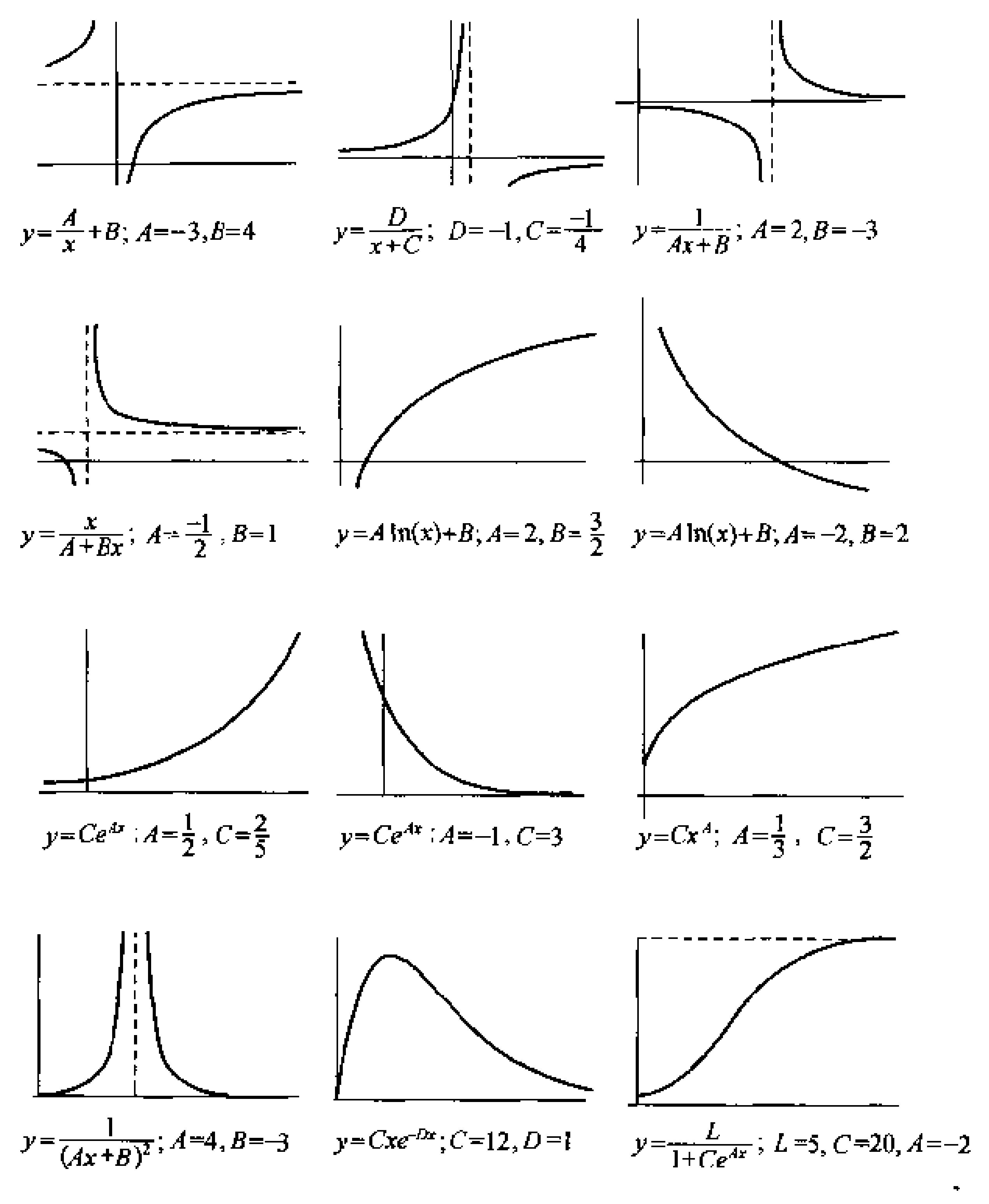


图 5.7　在“数据线性化”中可能使用的曲线

5.2.4　线性最小二乘法

设有 N 个数据点 $\{(x_k, y_k)\}$，并给定 M 个线性独立函数 $\{f_j(x)\}$。为求 M 个系数，使用由线性组合形成的函数 $f(x)$，表示为：

$$f(x) = \sum_{j=1}^{M} c_j f_j(x) \tag{18}$$

求解最小误差平方和，表示为：

$$E(c_1, c_2, \cdots, c_M) = \sum_{k=1}^{N} (f(x_k) - y_k)^2 = \sum_{k=1}^{N} \left(\left(\sum_{j=1}^{M} c_j f_j(x_k) \right) - y_k \right)^2 \tag{19}$$

为求解 E 的最小值，每个偏导数必须为零(即 $\partial E/\partial c_i = 0, i = 1, 2, \cdots, M$)，这样可得到如下方程组：

$$\sum_{k=1}^{N} \left(\left(\sum_{j=1}^{M} c_j f_j(x_k) \right) - y_k \right) (f_i(x_k)) = 0 \quad , \quad i = 1, 2, \cdots, M \tag{20}$$

交换式(20)中的求和顺序，可得一个 $M \times M$ 线性方程组，未知数是系数 $\{c_j\}$，称为正规方

程：

$$\sum_{j=1}^{M}\left(\sum_{k=1}^{N}f_j(x_k)f_i(x_k)\right)c_j = \sum_{k=1}^{N}f_i(x_k)y_k, \qquad i = 1,2,\cdots,M \tag{21}$$

5.2.5 矩阵公式

尽管(21)是一个有 M 个未知数的 M 阶线性方程组，但将它表示成矩阵形式可减少不必要的计算量。关键是构造如下矩阵 $\boldsymbol{F}$ 和 $\boldsymbol{F}'$：

$$\boldsymbol{F}=\begin{bmatrix} f_1(x_1) & f_2(x_1) & \cdots & f_M(x_1)\\ f_1(x_2) & f_2(x_2) & \cdots & f_M(x_2)\\ f_1(x_3) & f_2(x_3) & \cdots & f_M(x_3)\\ \vdots & \vdots & & \vdots\\ f_1(x_N) & f_2(x_N) & \cdots & f_M(x_N)\end{bmatrix}$$

$$\boldsymbol{F}'=\begin{bmatrix} f_1(x_1) & f_1(x_2) & f_1(x_3) & \cdots & f_1(x_N)\\ f_2(x_1) & f_2(x_2) & f_2(x_3) & \cdots & f_2(x_N)\\ \vdots & \vdots & \vdots & & \vdots\\ f_M(x_1) & f_M(x_2) & f_M(x_3) & \cdots & f_M(x_N)\end{bmatrix}$$

设 $\boldsymbol{F}'$和列矩阵 $\boldsymbol{Y}$ 的乘积表示为：

$$\boldsymbol{F}'\boldsymbol{Y}=\begin{bmatrix} f_1(x_1) & f_1(x_2) & f_1(x_3) & \cdots & f_1(x_N)\\ f_2(x_1) & f_2(x_2) & f_2(x_3) & \cdots & f_2(x_N)\\ \vdots & \vdots & \vdots & & \vdots\\ f_M(x_1) & f_M(x_2) & f_M(x_3) & \cdots & f_M(x_N)\end{bmatrix}\begin{bmatrix} y_1\\ y_2\\ \vdots\\ y_N\end{bmatrix} \tag{22}$$

式(22)中乘积 $\boldsymbol{F}'\boldsymbol{Y}$ 的行 i 的元素与式(21)中列矩阵的第 i 个元素相同，即：

$$\sum_{k=1}^{N}f_i(x_k)y_k = \text{row}_i\boldsymbol{F}'\cdot[y_1 \quad y_2 \quad \cdots \quad y_N]' \tag{23}$$

乘积 $\boldsymbol{F}'\boldsymbol{F}$，是一个 $M\times M$ 矩阵：

$$\boldsymbol{F}'\boldsymbol{F}=\begin{bmatrix} f_1(x_1) & f_1(x_2) & f_1(x_3) & \cdots & f_1(x_N)\\ f_2(x_1) & f_2(x_2) & f_2(x_3) & \cdots & f_2(x_N)\\ \vdots & \vdots & \vdots & & \vdots\\ f_M(x_1) & f_M(x_2) & f_M(x_3) & \cdots & f_M(x_N)\end{bmatrix}\begin{bmatrix} f_1(x_1) & f_2(x_1) & \cdots & f_M(x_1)\\ f_1(x_2) & f_2(x_2) & \cdots & f_M(x_2)\\ f_1(x_3) & f_2(x_3) & \cdots & f_M(x_3)\\ \vdots & \vdots & & \vdots\\ f_1(x_N) & f_2(x_N) & \cdots & f_M(x_N)\end{bmatrix}$$

$\boldsymbol{F}'\boldsymbol{F}$ 中位于第 i 行和第 j 列的元素是式(21)中第 i 行的系数 c_j，即：

$$\sum_{k=1}^{N}f_i(x_k)f_j(x_k) = f_i(x_1)f_j(x_1) + f_i(x_2)f_j(x_2) + \cdots + f_i(x_N)f_j(x_N) \tag{24}$$

当 M 很小时，有效计算式(18)中最小二乘系数的一种方式是存储矩阵 $\boldsymbol{F}$，计算 $\boldsymbol{F}'\boldsymbol{F}$ 和 $\boldsymbol{F}'\boldsymbol{Y}$，然后求解线性方程组：

$$\boldsymbol{F}'\boldsymbol{F}\boldsymbol{C} = \boldsymbol{F}'\boldsymbol{Y}(\text{求解系数矩阵 }\boldsymbol{C}) \tag{25}$$

5.2.6 多项式拟合

当采用前述的方法使用函数集合 $\{f_j(x) = x^{j-1}\}$，索引值范围从 $j=1$ 到 $j=M+1$ 时，函数

$f(x)$为 M 阶多项式：

$$f(x)=c_1+c_2x+c_3x^2+\cdots+c_{M+1}x^M \tag{26}$$

现在考虑如何求解最小二乘抛物线拟合，而把扩展到更高维的情况留给读者研究。

定理 5.3(最小二乘抛物线拟合) 设$\{(x_k,y_k)\}_{k=1}^{N}$有 N 个点，横坐标是确定的。最小二乘抛物线的系数表示为：

$$y=f(x)=Ax^2+Bx+C \tag{27}$$

求解 A,B,C 的线性方程组为：

$$\begin{aligned}
\left(\sum_{k=1}^{N}x_k^4\right)A+\left(\sum_{k=1}^{N}x_k^3\right)B+\left(\sum_{k=1}^{N}x_k^2\right)C&=\sum_{k=1}^{N}y_kx_k^2\\
\left(\sum_{k=1}^{N}x_k^3\right)A+\left(\sum_{k=1}^{N}x_k^2\right)B+\left(\sum_{k=1}^{N}x_k\right)C&=\sum_{k=1}^{N}y_kx_k\\
\left(\sum_{k=1}^{N}x_k^2\right)A+\left(\sum_{k=1}^{N}x_k\right)B+NC&=\sum_{k=1}^{N}y_k
\end{aligned} \tag{28}$$

证明：通过求下面表达式的最小值可得到 A,B,C：

$$E(A,B,C)=\sum_{k=1}^{N}(Ax_k^2+Bx_k+C-y_k)^2 \tag{29}$$

令偏导数 $\partial E/\partial A,\partial E/\partial B,\partial E/\partial C$ 为零，可得：

$$\begin{aligned}
0&=\frac{\partial E(A,B,C)}{\partial A}=2\sum_{k=1}^{N}(Ax_k^2+Bx_k+C-y_k)^1(x_k^2)\\
0&=\frac{\partial E(A,B,C)}{\partial B}=2\sum_{k=1}^{N}(Ax_k^2+Bx_k+C-y_k)^1(x_k)\\
0&=\frac{\partial E(A,B,C)}{\partial C}=2\sum_{k=1}^{N}(Ax_k^2+Bx_k+C-y_k)^1(1)
\end{aligned} \tag{30}$$

利用加法分配律，可将式(30)中的 A,B,C 移到求和的外面，因此得到式(28)中的正规方程。

表 5.7 求解例 5.6 中最小二乘抛物线的系数

x_k	y_k	x_k^2	x_k^3	x_k^4	x_ky_k	$x_k^2y_k$
-3	3	9	-27	81	-9	27
0	1	0	0	0	0	0
2	1	4	8	16	2	4
4	3	16	64	256	12	48
3	8	29	45	353	5	79

例 5.6 根据 4 个数据点$(-3,3),(0,1),(2,1),(4,3)$，求解最小二乘抛物线。

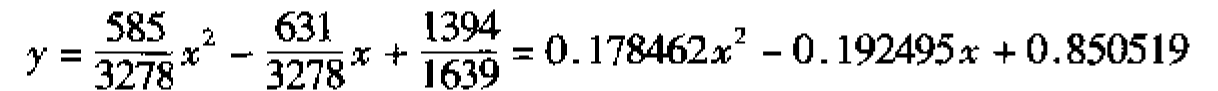
解：

可用表 5.7 中的项计算线性方程组(28)中的求和计算。

求解 A、B、C 的线性方程组(28)表示为：

$$\begin{aligned}
353A+45B+29C&=79\\
45A+29B+3C&=5\\
29A+3B+4C&=8
\end{aligned}$$

线性方程组的解为 $A=585/3278,B=-631/3278,C=1394/1639$，抛物线(如图 5.8 所示)为：



$$y = \frac{585}{3278}x^2 - \frac{631}{3278}x + \frac{1394}{1639} = 0.178462x^2 - 0.192495x + 0.850519$$

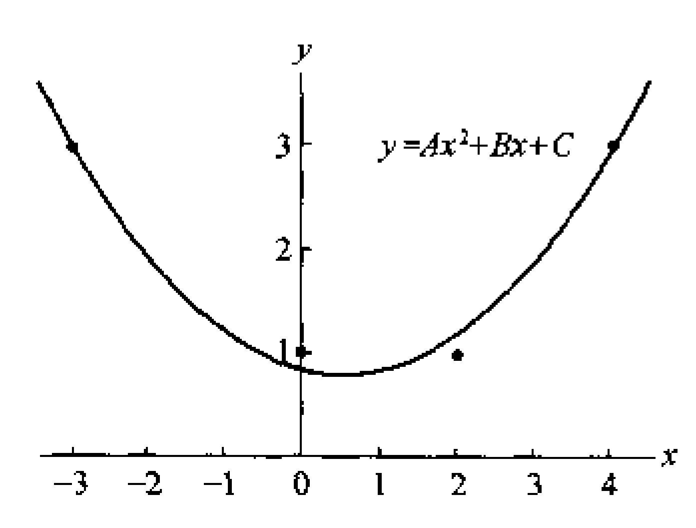


图 5.8 例 5.6 中的最小二乘抛物线

5.2.7 多项式摆动

使用最小二乘多项式拟合非线性数据的方法具有吸引力。但如果数据不具有多项式特性，则求出的曲线可能产生大的振荡。这种现象称为多项式摆动，它在高阶多项式情况下更容易发生。由于这个原因，一般很少使用超过 6 阶的多项式，除非已知所使用的多项式是真实的多项式。

例如，用函数 $f(x) = 1.44/x^2 + 0.24x$ 生成 6 个数据点(0.25, 23.1), (1.0, 1.68), (1.5, 1.0), (2.0, 0.84), (2.4, 0.826), (5.0, 1.2576)。使用曲线拟合得到的最小二乘多项式为：

$$P_2(x) = 22.93 - 16.96x + 2.553x^2$$

$$P_3(x) = 33.04 - 46.51x + 19.51x^2 - 2.296x^3$$

$$P_4(x) = 39.92 - 80.93x + 58.39x^2 - 17.15x^3 + 1.680x^4$$

和：

$$P_5(x) = 46.02 - 118.1x + 119.4x^2 - 57.51x^3 + 13.03x^4 - 1.085x^5$$

这些多项式如图 5.9(a)到(d)所示。注意 $P_3(x)$, $P_4(x)$, $P_5(x)$在区间[2,5]内有较大的摆动。尽管 $P_5(x)$经过 6 个点，但它的拟合最差。如果必须使用多项式拟合这些数据，则$P_2(x)$是较好的选择。

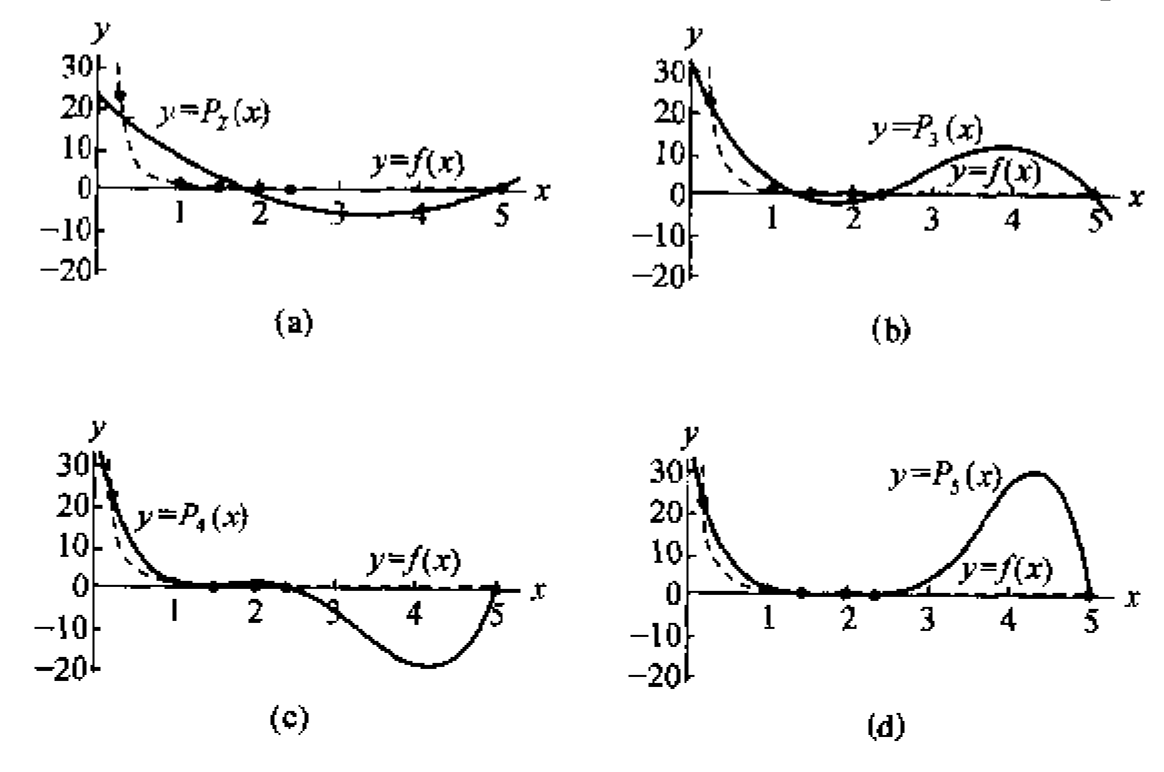


图 5.9 (a)使用 $P_2(x)$拟合数据；(b)使用 $P_3(x)$拟合数据
(c)使用 $P_4(x)$拟合数据；(d)使用 $P_5(x)$拟合数据

下面的程序使用矩阵 F,它包含式(18)中的项 $f_j(x_k)=x_k^{j-1}$。

程序 5.2 (最小二乘多项式拟合)构造 M 阶最小二乘多项式:

$$P_M(x)=c_1+c_2x+c_3x^2+\cdots+c_Mx^{M-1}+c_{M+1}x^M$$

拟合 N 个数据点 $\{(x_k,y_k)\}_{k=1}^N$

```
function C = lspoly(X,Y,M)
% Input    - X is the 1xn abscissa vector
%          - Y is the 1xn ordinate vector
%          - M is the degree of the least-squares polynomial
% Output   - C is the coefficient list for the polynomial
n = length(X);
B = zeros(1:M+1);
F = zeros(n,M+1);
% Fifll the columns of F with the powers of X
for k = 1:M+1
  F(:,k) = x'.^(k-1);
end
% Solve the linear system from (25)
A = F' * F;
B = F' * Y;
C = A \ B;
C = flipud(c);
```

5.2.8 曲线拟合的练习

1. 对下列数据集,求解最小二乘抛物线 $f(x)=Ax^2+Bx+C$。

(a)

x_k	y_k
-3	15
-1	5
1	1
3	5

(b)

x_k	y_k
-3	-1
-1	25
1	25
3	1

2. 对下列数据集,求解最小二乘抛物线 $f(x)=Ax^2+Bx+C$。

(a)

x_k	y_k
-2	-5.8
-1	1.1
0	3.8
1	3.3
2	-1.5

(b)

x_k	y_k
-2	2.8
-1	2.1
0	3.25
1	6.0
2	11.5

(c)

x_k	y_k
-2	10
-1	1
0	0
1	2
2	9

3. 对给定的数据集,求解最小二乘曲线:

(a) $f(x)=Ce^{Ax}$,使用表 5.6 中的变量变换 $X=x,Y=\ln(y)$和 $C=e^B$ 来线性化数据点。

(b) $f(x)=Cx^A$,使用表 5.6 中的变量变换 $X=\ln(x),Y=\ln(y)$和 $C=e^B$ 来线性化数据点。

(c) 使用 $E_2(f)$判断哪个曲线是最佳拟合。

给定数据集为：

x_k	y_k
1	0.6
2	1.9
3	4.3
4	7.6
5	12.6

4. 对下面给定的数据集，求解最小二乘曲线：
   (a) $f(x)=Ce^{Ax}$，使用表5.6中的变量变换 $X=x$，$Y=\ln(y)$和 $C=e^B$ 线性化数据点。
   (b) $f(x)=1/(Ax+B)$，使用表5.6中的变量变换 $X=x$ 和 $Y=1/y$ 线性化数据点。
   (c) 使用 $E_2(f)$判断哪个曲线是最佳拟合。
   给定数据集为：

x_k	y_k
-1	6.62
0	3.94
1	2.17
2	1.35
3	0.89

5. 对下面给定的数据集，求解最小二乘曲线：
   (a) $f(x)=Ce^{Ax}$，使用表5.6中的变量变换 $X=x$，$Y=\ln(y)$和 $C=e^B$ 线性化数据点。
   (b) $f(x)=(Ax+B)^{-2}$，使用表5.6中的变量变换 $X=x$ 和 $Y=y^{-1/2}$线性化数据点。
   (c) 使用 $E_2(f)$判断哪个曲线是最佳拟合。
   给定数据集为：

(i)

x_k	y_k
-1	13.45
0	3.01
1	0.67
2	0.15

(ii)

x_k	y_k
-1	13.65
0	1.38
1	0.49
3	0.15

6. 对数人口增长。当人口 $P(t)$受限于极值 L 时，它符合对数曲线，具有形式 $P(t)=L/(1+Ce^{At})$。对下列数据集求解系数 A 和 C，L 是已知的。
   (a) (0,200)，(1,400)，(2,650)，(3,850)，(4,950)和 $L=1000$。
   (b) (0,500)，(1,1000)，(2,1800)，(3,2800)，(4,3700)和 $L=5000$。
7. 利用美国人口数据，求解对数曲线 $P(t)$，并估计2000年时的美国人口。
   给定人口数据为：

(a)设 $L=8\times10^8$

年	t_k	P_k
1800	-10	5.3
1850	-5	23.2
1900	0	76.1
1950	5	152.3

(b)设 $L=8\times10^8$

年	t_k	P_k
1900	0	76.1
1920	2	106.5
1940	4	132.6
1960	6	180.7
1980	8	226.5

在练习 8 到练习 15 中,执行表 5.6 中的变量变换,并对下列每个函数求出线性化表示:

8. $y = \frac{A}{x} + B$

9. $y = \frac{D}{x + C}$

10. $y = \frac{1}{Ax + B}$

11. $y = \frac{x}{A + Bx}$

12. $y = A\ln(x) + B$

13. $y = Cx^A$

14. $y = (Ax + B)^{-2}$

15. $y = Cxe^{-Dx}$

16. (a) 根据定理5.3 中的证明过程,推导最小二乘曲线 $f(x) = A\cos(x) + B\sin(x)$的正规方程。

(b) 利用(a)的结果,对下列数据集求解最小二乘曲线 $f(x) = A\cos(x) + B\sin(x)$。给定数据集为:

x_k	y_k
-3.0	-0.1385
-1.5	-2.1587
0.0	0.8330
1.5	2.2774
3.0	-0.5110

17. 针对 N 个点 $z = Ax + By + C$ 的最小二乘平面$(x_1, y_1, z_1), \ldots, (x_N, y_N, z_N)$,可通过下式的最小化得到:

$$E(A, B, C) = \sum_{k=1}^{N}(Ax_k + By_k + C - z_k)^2$$

试推导正规方程:

$$\left(\sum_{k=1}^{N} x_k^2\right)A + \left(\sum_{k=1}^{N} x_k y_k\right)B + \left(\sum_{k=1}^{N} x_k\right)C = \sum_{k=1}^{N} z_k x_k$$

$$\left(\sum_{k=1}^{N} x_k y_k\right)A + \left(\sum_{k=1}^{N} y_k^2\right)B + \left(\sum_{k=1}^{N} y_k\right)C = \sum_{k=1}^{N} z_k x_k$$

$$\left(\sum_{k=1}^{N} x_k\right)A + \left(\sum_{k=1}^{N} y_k\right)B + NC = \sum_{k=1}^{N} z_k$$

18. 对下列数据集求解最小二乘平面:

(a) (1,1,7),(1,2,9),(2,1,10),(2,2,11),(2,3,12)

(b) (1,2,6),(2,3,7),(1,1,8),(2,2,8),(2,1,9)

(c) (3,1,-3),(2,1,-1),(2,2,0),(1,1,1),(1,2,3)

19. 设有如下数据表:

x_k	y_k
1.0	2.0
2.0	5.0
3.0	10.0
4.0	17.0
5.0	26.0

当对函数 $y = D/(x + C)$使用变量变换 $X = xy$ 和 $Y = 1/y$ 后,变换的最小二乘拟合为:

$$y = \frac{-17.719403}{x - 5.476617}$$

当对函数 $y = 1/(Ax + B)$ 使用变量变换 $X = x$ 和 $Y = 1/y$ 后，变换的最小二乘拟合为：

$$y = \frac{1}{-0.1064253x + 0.4987330}$$

判断哪个拟合是更好，说明为什么其中一个结果是不合理的。

5.2.9 算法和程序

1. 洛杉矶郊区在 11 月 8 日的温度记录如下表所示。共有 24 个数据点。

   (a) 根据例 5.5 中的处理过程（使用 fmins 命令），对给定的数据集求解最小二乘曲线 $f(x) = A\cos(Bx) + C\sin(Dx)$。

   (b) 求 $E_2(f)$。

   (c) 在同一坐标系下画出这些点集和(a)得出的最小二乘曲线。

   给定温度记录表如下：

时　间	温　度	时　间	温　度
1	66	1	58
2	66	2	58
3	65	3	58
4	64	4	58
5	63	5	57
6	63	6	57
7	62	7	57
8	61	8	58
9	60	9	60
10	60	10	64
11	59	11	67
午夜	58	正午	68

5.3 样条函数插值

对 $N+1$ 个点 $\{(x_k, y_k)\}_{k=0}^{N}$ 的多项式插值经常不令人满意。从 5.2 节可知，一个 N 阶多项式可能有 N－1 个相对极大值和极小值，同时曲线可能会摆动以经过这些点。另一个方法是将图形分段，每段为一个低阶多项式 $S_k(x)$，并在相邻点 (x_k, y_k) 和 (x_{k+1}, y_{k+1}) 之间进行插值（如图 5.10 所示）。

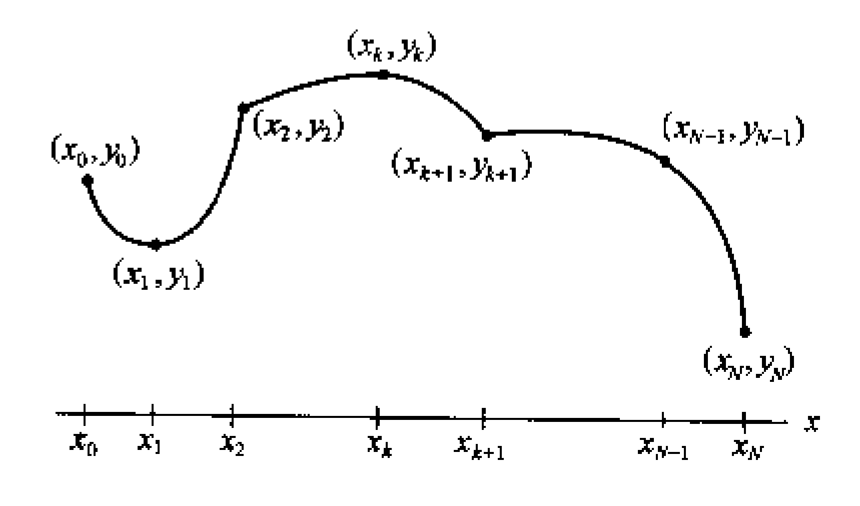


图 5.10　分段多项式插值

两个相邻的曲线部分 $y = S_k(x)$ 和 $y = S_{k+1}(x)$ 分别位于区间 $[x_k, x_{k+1}]$ 和 $[x_{k+1}, x_{k+2}]$，并经过共同的结点(knot)(x_{k+1}, y_{k+1})。曲线的这两部分在结点 (x_{k+1}, y_{k+1}) 处连接在一起，而且函数集合 $\{S_k(x)\}$ 形成一个分段多项式曲线，表示为 $S(x)$。

5.3.1 分段线性插值

最简单的多项式是一阶多项式，即经过各点的多项式路径是由包含各点的直线段组成。4.3 节的拉格朗日多项式可用来表示分段线性曲线：

$$S_k(x) = y_k \frac{x - x_{k+1}}{x_k - x_{k+1}} + y_{k+1} \frac{x - x_k}{x_{k+1} - x_k}, \qquad x_k \leqslant x \leqslant x_{k+1} \tag{1}$$

得到的曲线看起来像折线(如图 5.11 所示)。

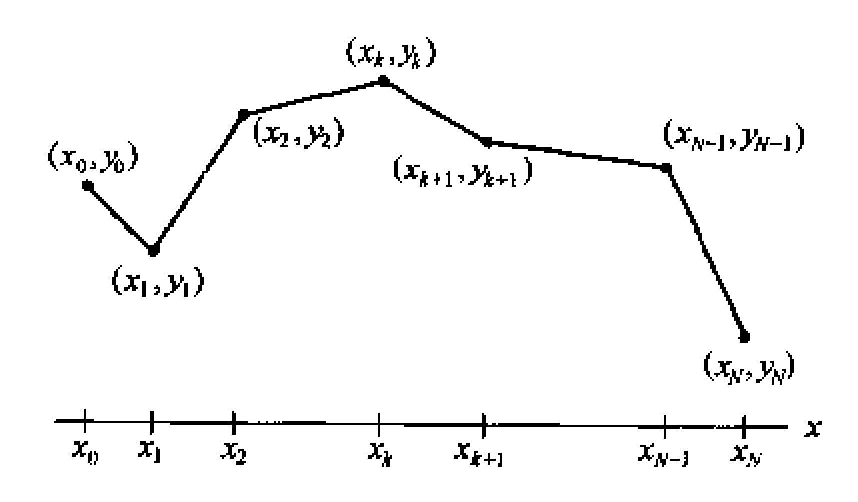


图 5.11 分段线性插值(线性样条曲线)

如果利用线段点的斜率公式，可得到一个等价的表达式：

$$S_k(x) = y_k + d_k(x - x_k)$$

其中 $d_k = (y_{k+1} - y_k)/(x_{k+1} - x_k)$。得到的线性样条函数可表示为如下形式：

$$S(x) = \begin{cases} y_0 + d_0(x - x_0) & x \text{ 在}[x_0, x_1]\text{内} \\ y_1 + d_1(x - x_1) & x \text{ 在}[x_1, x_2]\text{内} \\ \vdots & \vdots \\ y_k + d_k(x - x_k) & x \text{ 在}[x_k, x_{k+1}]\text{内} \\ \vdots & \vdots \\ y_{N-1} + d_{N-1}(x - x_{N-1}) & x \text{ 在}[x_{N-1}, x_N]\text{内} \end{cases} \tag{2}$$

对于显式计算 $S(x)$，式(2)的形式优于式(1)。这里假设横坐标按 $x_0 < x_1 < \cdots < x_{N-1} < x_N$ 排序。对于一个固定值 x，通过连续计算差值 $x - x_1, \cdots, x - x_k, x - x_{k+1}$ 直到 $k+1$ 是满足 $x - x_{k+1} < 0$ 的最小整数，可找到包含 x 的区间 $[x_k, x_{k+1}]$。因此可找到 k，满足 $x_k \leqslant x \leqslant x_{k+1}$，且样条函数 $S(x)$ 的值表示为：

$$S(x) = S_k(x) = y_k + d_k(x - x_k), \quad x_k \leqslant x \leqslant x_{k+1} \tag{3}$$

这些技术可扩展到更高阶的多项式。例如，如果给定奇数个点 $x_0, x_1, \cdots, x_{2M}$，则可对每个子区间 $[x_{2k}, x_{2k+2}]$ $(k = 0, 1, \cdots, M-1)$，构造一个分段二次多项式。采用二次多项式样条的一个缺点是在偶数点 x_{2k} 处的曲率变化很大，这可能导致曲线出现非期望的弯曲或畸变。二次样条曲线的二阶导数在偶数点不连续。如果利用分段三次多项式，则一阶导数和二阶导数都连续。

5.3.2 分段三次样条曲线

数据集进行多项式曲线拟合在CAD、CAM和计算机图形系统中有许多应用。操作者希望画出没有误差的经过数据点的光滑曲线。传统上,一般使用曲线板或设计师的样条主观地画出曲线,只要目测时觉得光滑就行。从数学角度上分析,在每个区间$[x_k, x_{k+1}]$可构造一个三次函数,使得分段曲线$y=S(x)$和它的一阶导数和二阶导数在更大的区间$[x_0, x_N]$内连续。$S'(x)$的连续性意味着曲线$y=S(x)$没有急弯。$S''(x)$的连续性意味着每点的曲率半径有定义。

定义5.1(三次样条插值) 设$\{(x_k, y_k)\}_{k=0}^{N}$有$N+1$个点,其中$a=x_0<x_1<\cdots<x_N=b$。如果存在N个三次多项式$S_k(x)$,系数为$s_{k,0}, s_{k,1}, s_{k,2}, s_{k,3}$,满足如下性质:

Ⅰ. $S(x)=S_k(x)=s_{k,0}+s_{k,1}(x-x_k)+s_{k,2}(x-x_k)^2+s_{k,3}(x-x_k)^3$,
$x\in[x_k, x_{k+1}]$且$k=0,1,\cdots,N-1$

Ⅱ. $S(x_k)=y_k$, $k=0,1,\cdots,N$

Ⅲ. $S_k(x_{k+1})=S_{k+1}(x_{k+1})$, $k=0,1,\cdots,N-2$

Ⅳ. $S'_k(x_{k+1})=S'_{k+1}(x_{k+1})$, $k=0,1,\cdots,N-2$

Ⅴ. $S''_k(x_{k+1})=S''_{k+1}(x_{k+1})$, $k=0,1,\cdots,N-2$

则称函数$S(x)$为三次样条函数。

性质Ⅰ描述了由分段三次多项式构成的$S(x)$。性质Ⅱ描述了对给定数据点集的分段三次插值。性质Ⅲ和性质Ⅳ是保证分段三次多项式函数是一个光滑连续函数。性质Ⅴ保证了函数的二阶导数也是连续的。

5.3.3 三次样条的存在性

是否可能构造一个三次样条满足性质Ⅰ到性质Ⅴ?每个三次多项式$S_k(x)$有四个未知常数($s_{k,0}, s_{k,1}, s_{k,2}$和$s_{k,3}$),因此需要求解$4N$个系数。换言之,要确定$4N$个自由度或条件,数据点提供了$N+1$个条件,性质Ⅲ、性质Ⅳ和性质Ⅴ每个提供了$N-1$个条件,总共确定了$N+1+3(N-1)=4N-2$个条件。这样剩下了两个自由度。可称之为端点约束:它们包括在点x_0和x_N处的导数$S'(x)$或$S''(x)$,在下面将会讨论。现在构造三次样条。

由于$S(x)$是分段三次多项式,它的二阶导数$S''(x)$在区间$[x_0, x_N]$内是分段线性的。根据线性拉格朗日插值,$S''(x)=S''_k(x)$可表示为:

$$S''_k(x)=S''(x_k)\frac{x-x_{k+1}}{x_k-x_{k+1}}+S''(x_{k+1})\frac{x-x_k}{x_{k+1}-x_k} \tag{4}$$

用$m_k=S''(x_k)$, $m_{k+1}=S''(x_{k+1})$和$h_k=x_{k+1}-x_k$可得:

$$S''_k(x)=\frac{m_k}{h_k}(x_{k+1}-x)+\frac{m_{k+1}}{h_k}(x-x_k) \tag{5}$$

其中$x_k\leqslant x\leqslant x_{k+1}$, $k=0,1,\cdots,N-1$。将式(5)积分两次,会引入两个积分常数,并得到如下形式:

$$S_k(x)=\frac{m_k}{6h_k}(x_{k+1}-x)^3+\frac{m_{k+1}}{6h_k}(x-x_k)^3+p_k(x_{k+1}-x)+q_k(x-x_k) \tag{6}$$

将 x_k 和 x_k+1 代入式(6)中,并使用值 $y_k=S_k(x_k)$ 和 $y_{k+1}=S_k(x_{k+1})$ 可分别得到包含 p_k 和 q_k 的方程:

$$y_k=\frac{m_k}{6}h_k^2+p_kh_k \text{ 和 } y_{k+1}=\frac{m_{k+1}}{6}h_k^2+q_kh_k \tag{7}$$

求解这两个方程可容易得出 p_k 和 q_k,而且将这些值代入式(6)中,可得到如下的三次多项式方程:

$$\begin{aligned}S_k(x)=&-\frac{m_k}{6h_k}(x_{k+1}-x)^3+\frac{m_{k+1}}{6h_k}(x-x_k)^3\\&+\left(\frac{y_k}{h_k}-\frac{m_kh_k}{6}\right)(x_{k+1}-x)+\left(\frac{y_{k+1}}{h_k}-\frac{m_{k+1}h_k}{6}\right)(x-x_k)\end{aligned} \tag{8}$$

需要注意表达式(8)可简化为只包含未知系数$\{m_k\}$的形式。为求解这些值,必须使用式(8)的导数,即:

$$\begin{aligned}S'_k(x)=&-\frac{m_k}{2h_k}(x_{k+1}-x)^2+\frac{m_{k+1}}{2h_k}(x-x_k)^2\\&-\left(\frac{y_k}{h_k}-\frac{m_kh_k}{6}\right)+\frac{y_{k+1}}{h_k}-\frac{m_{k+1}h_k}{h_k}\end{aligned} \tag{9}$$

在 x_k 处计算式(9),并简化结果可得到:

$$S'_k(x_k)=-\frac{m_k}{3}h_k-\frac{m_{k+1}}{6}h_k+d_k,\quad d_k=\frac{y_{k+1}-y_k}{h_k} \tag{10}$$

同理,在式(9)中用 $k-1$ 代替 k 得到 $S'_{k-1}(x)$,并计算在 x_k 处的解为:

$$S'_{k-1}(x_k)=\frac{m_k}{3}h_{k-1}+\frac{m_{k-1}}{6}h_{k-1}+d_{k-1} \tag{11}$$

现在利用性质Ⅳ和式(10)和式(11)得到包含 m_{k-1}, m_k 和 m_{k+1} 的重要关系式:

$$h_{k-1}m_{k-1}+2(h_{k-1}+h_k)m_k+h_km_{k+1}=u_k \tag{12}$$

其中 $u_k=6(d_k-d_{k-1})$, $k=1,2,\cdots,N-1$。

5.3.4 构造三次样条

式(12)中的未知数是要求的值$\{m_k\}$,而且其他的项可通过数据点集$\{(x_k,y_k)\}$进行简单的数学计算得到的常量。因此,式(12)是一个包含 $N+1$ 个未知数、具有 $N-1$ 个线性方程的不定方程组。所以需要另外两个方程组才能求解。可通过它们消去第一个方程中的 m_0 和第 $N-1$ 个方程中的 m_N。针对端点约束的标准策略如表 5.8 所示。

表 5.8 针对三次样条的端点约束

策略描述	包含 m_0 和 m_N 的方程
(i)压紧三次样条,确定 $S'(x)$, $S'(x_n)$ (如果导数已知,这是"最佳选择")	$m_0=\frac{3}{h_0}(d_0-S'(x_0))-\frac{m_1}{2}$ $m_N=\frac{3}{h_{N-1}}(S'(x_N)-d_{N-1})-\frac{m_{N-1}}{2}$

（续表）

策略描述	包含 m_0 和 m_N 的方程
(ii)Natural 三次样条(一个"松驰曲线")	$m_0=0, m_N=0$
(iii)外推 $S''(x)$到端点	$m_0=m_1-\frac{h_0(m_2-m_1)}{h_1}$ $m_N=m_{N-1}+\frac{h_{N-1}(m_{N-1}-m_{N-2})}{h_{N-2}}$
(iv)$S''(x)$是靠近端点的常量	$m_0=m_1, m_N=m_{N-1}$
(v)在每个端点处确定 $S''(x)$	$m_0=S''(x_0), m_N=S''(x_N)$

设采用表 5.8 中的策略(v)。如果给定 m_0，则可计算出 h_0m_0，而且式(12)的第一个方程（当 $k=1$ 时）为：

$$2(h_0+h_1)m_1+h_1m_2=u_1-h_0m_0 \tag{13}$$

同理，如果给定 m_N，则可计算出 $h_{N-1}m_N$，而且式(12)的最后一个方程（当 $k=N-1$ 时）为：

$$h_{N-2}m_{N-2}+2(h_{N-2}+h_{N-1})m_{N-1}=u_{N-1}-h_{N-1}m_N \tag{14}$$

式(13)和式(14)结合式(12)，其中 $k=2,3,\cdots,N-2$，可形成包含系数 $m_1,m_2,\cdots,m_{N-1}$ 的 $N-1$ 阶线性方程组。

如果不管在表 5.8 中选择的特定策略，可重写式(12)中的方程 1 到方程 $N-1$，得到一个包含 $m_1,m_2,\cdots,m_{N-1}$ 的三角线性方程组 $\boldsymbol{HM}=\boldsymbol{V}$，表示为：

$$\begin{bmatrix} b_1 & c_1 & & & & \\ a_1 & b_2 & c_2 & & & \\ & & \ddots & & & \\ & & & a_{N-3} & b_{N-2} & c_{N-2} \\ & & & & a_{N-2} & b_{N-1} \end{bmatrix}\begin{bmatrix} m_1 \\ m_2 \\ \vdots \\ m_{N-2} \\ m_{N-1} \end{bmatrix}=\begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_{N-2} \\ v_{N-1} \end{bmatrix} \tag{15}$$

式(15)中的线性方程组具有严格对角优势，并有惟一解（细节参见第 3 章）。当得到系数 $\{m_k\}$ 后，可利用如下公式计算 $S_k(x)$ 的样条系数 $\{s_{k,j}\}$：

$$s_{k,0}=y_k \qquad S_{k,1}=d_k-\frac{h_k(2m_k+m_{k+1})}{6}$$
$$s_{k,2}=\frac{m_k}{2} \qquad s_{k,3}=\frac{m_{k+1}-m_k}{6h_k} \tag{16}$$

为了更有效地计算，每个三次多项式 $S_k(x)$ 可表示成嵌套乘的形式：

$$S_k(x)=((S_{k,3}w+S_{k,2})w+S_{k,1})w+y_k, w=x-x_k \tag{17}$$

这里 $S_k(x)$ 在区间 $x_k\leqslant x\leqslant x_{k+1}$ 内使用。

结合表 5.8 中策略的式(12)可用来构造在端点处有特殊性质的三次样条。特别是表 5.8 中的 m_0 和 m_N 可用来定制式(12)中的第一个和最后一个方程，并形成式(15)中的 $N-1$ 阶方程组。然后可求解对角方程组的系数 $m_1,m_2,\cdots,m_{N-1}$。最后用式(16)中的公式求出样条系数。下面指出了对不同类型的样条如何构造方程组。

5.3.5 端点约束

下面的 5 个引理说明了对表 5.8 中的不同端点约束,要求解的三角线性方程组的形式。

引理 5.1(压紧(Clamped)样条) 存在惟一的三次样条曲线,其一阶导数的边界条件是 $S'(a) = d_0$ 和 $S'(b) = d_N$。

证明:求解下列线性方程组:

$$\left(\frac{3}{2}h_0 + 2h_1\right)m_1 + h_1 m_2 = u_1 - 3(d_0 - S'(x_0))$$

$$h_{k-1}m_{k-1} + 2(h_{k-1} + h_k)m_k + h_k m_{k+1} = u_k, \qquad k = 2,3,\cdots,N-2$$

$$h_{N-2}m_{N-2} + (2h_{N-2} + \frac{3}{2}h_{N-1})m_{N-1} = u_{N-1} - 3(S'(x_N) - d_{N-1})$$

注:压紧样条在端点有斜率。从可视化的角度分析,当柔软有弹性的木杆用外力使其经过数据点,而且木杆在端点处被夹紧时,得到的曲线为压紧样条。这样的样条能够为需要绘制经过多个点光滑曲线的绘图员提供帮助。

引理 5.2(自然(Natural)样条) 存在惟一的三次样条曲线,它的自由边界条件是 $S''(a)=0$ 和 $S''(b)=0$。

证明:求解下列线性方程组:

$$2(h_0 + h_1)m_1 + h_1 m_2 = u_1$$

$$h_{k-1}m_{k-1} + 2(h_{k-1} + h_k)m_k + h_k m_{k+1} = u_k, \qquad k = 2,3,\cdots,N-2$$

$$h_{N-2}m_{N-2} + 2(h_{N-2} + h_{N-1})m_{N-1} = u_{N-1}$$

注:自然样条是柔软有弹性的木杆经过所有数据点形成的曲线,但让端点的斜率自由地在某一位置保持平衡,使得曲线的摇摆行为最小。它对有多位有效数字精度的试验数据进行曲线拟合时很有用。

引理 5.3(外推(Extrapolated)样条) 存在惟一的三次样条曲线,其中通过对点 x_1 和 x_2 进行外推得到 $S''(a)$,同时通过对点 x_{N-1} 和 x_{N-2} 进行外推得到 $S''(b)$。

证明:求解下列线性方程组:

$$\left(3h_0 2h_1 + \frac{h_0^2}{h_1}\right)m_1 + \left(h_1 - \frac{h_0^2}{h_1}\right)m_2 = u_1$$

$$h_{k-1}m_{k-1} + 2(h_{k-1} + h_k)m_k + h_k m_{k+1} = u_k, \qquad k = 2,3,\cdots,N-2$$

$$\left(h_{N-2} - \frac{h_{N-1}^2}{h_{N-2}}\right)m_{N-2} + \left(2h_{N-2} + 3h_{N-1} + \frac{h_{N-1}^2}{h_{N-2}}\right)m_{N-1} = u_{N-1}$$

注:外推样条等价于端点处的三次多项式曲线是相邻三次多项式曲线的扩展形成的样条,也就是说,样条曲线在区间$[x_0,x_2]$内形成单个三次多项式曲线,同时在区间$[x_{N-2},x_N]$内形成另一个三次多项式曲线。

引理 5.4(抛物线终结(Parabolically Terminated)样条) 存在惟一的三次样条曲线,其中在区间$[x_0,x_1]$内 $S'''\equiv 0$,而在区间$[x_{N-1},x_N]$内 $S'''\equiv 0$。

证明：求解下列线性方程组：

$$(3h_0 + 2h_1)m_1 + h_1 m_2 = u_1$$
$$h_{k-1}m_{k-1} + 2(h_{k-1} + h_k)m_k + h_k m_{k+1} = u_k, \qquad k = 2,3,\cdots,N-2$$
$$h_{N-2}m_{N-2} + (2h_{N-2} + 3h_{N-1})m_{N-1} = u_{N-1}$$

注：在区间$[x_0,x_1]$内 $S''(x)\equiv 0$，使得三次多项式曲线退化为二次多项式曲线，同时在区间$[x_{N-1},x_N]$内也发生了同样的情况。

引理 5.5(端点曲率调整(End－point Curvature－adjusted)样条) 存在惟一的三次样条曲线，其中二阶导数的边界条件 $S''(a)$和 $S''(b)$是确定的。

证明：求解下列线性方程组：

$$2(h_0 + h_1)m_1 + h_1 m_2 = u_1 - h_0 S''(x_0)$$
$$h_{k-1}m_{k-1} + 2(h_{k-1} + h_k)m_k + h_k m_{k+1} = u_k, \qquad k = 2,3,\cdots,N-2$$
$$h_{N-2}m_{N-2} + 2(h_{N-2} + h_{N-1})m_{N-1} = u_{N-1} - h_{N-1}S''(x_N)$$

注：通过对 $S''(a)$和 $S''(b)$赋值，可调整每个端点的曲率。

下面的5个例子显示了不同样条的行为。通过混合端点条件可得到更多的样条形式，这留给读者进行研究。

例5.7 求压紧三次样条曲线，经过点(0,0),(1,0.5),(2,2.0),(3,1.5)，而且一阶导数的边界条件为 $S'(0)=0.2$ 和 $S'(3)=-1$。

解：

首先计算下面的值：

$$h_0 = h_1 = h_2 = 1$$
$$d_0 = (y_1 - y_0)/h_0 = (0.5-0.0)/1 = 0.5$$
$$d_1 = (y_2 - y_1)/h_1 = (2.0-0.5)/1 = 1.5$$
$$d_2 = (y_3 - y_2)/h_2 = (1.5-2.0)/1 = -0.5$$
$$u_1 = 6(d_1 - d_0) = 6(1.5-0.5) = 6.0$$
$$u_2 = 6(d_2 - d_1) = 6(-0.5-1.5) = -12.0$$

然后利用引理5.1得到如下方程组：

$$\left(\frac{3}{2}+2\right)m_1 + m_2 = 6.0 - 3(0.5-0.2) = 5.1$$

$$m_1 + \left(2+\frac{3}{2}\right)m_2 = -12.0 - 3(-1.0-(-0.5)) = -10.5$$

将上述方程组进行简化并表示成矩阵形式，可得到：

$$\begin{bmatrix} 3.5 & 1.0 \\ 1.0 & 3.5 \end{bmatrix}\begin{bmatrix} m_1 \\ m_2 \end{bmatrix} = \begin{bmatrix} 5.1 \\ -10.5 \end{bmatrix}$$

直接通过计算可得到方程组的解 $m_1 = 2.25$ 和 $m_2 = -3.72$。现在将它们代入表5.8的(i)中的方程组，以求解系数 m_0 和 m_3：

$$m_0 = 3(0.5 - 0.2) - \frac{2.52}{2} = -0.36$$

$$m_3 = 3(-1.0 + 0.5) - \frac{-3.72}{2} = 0.36$$

接下来,可求出 $m_0 = -0.36, m_1 = 2.25, m_2 = -3.72, m_3 = 0.36$,并将它们代入式(16)求解样条系数。结果为:

$$\begin{aligned} S_0(x) &= 0.48x^3 - 0.18x^2 + 0.2x, & 0 \leqslant x \leqslant 1 \\ S_1(x) &= -1.04(x-1)^3 + 1.26(x-1)^2 + 1.28(x-1) + 0.5, & 1 \leqslant x \leqslant 2 \\ S_2(x) &= 0.68(x-2)^3 - 1.86(x-2)^2 + 0.68(x-2) + 2.0, & 2 \leqslant x \leqslant 3 \end{aligned} \tag{18}$$

压紧三次样条如图 5.12 所示。

例 5.8 求自然三次样条曲线,经过点(0,0.0),(1,0.5),(2,2.0),(3,1.5),且自由边界条件为 $S''(x) = 0$ 和 $S''(3) = 0$。

使用例 5.7 中计算出的值$\{h_k\}$,$\{d_k\}$,$\{u_k\}$,根据引理 5.2 可得方程:

$$2(1+1)m_1 + m_2 = 6.0$$

$$m_1 + 2(1+1)m_2 = -12.0$$

线性方程组的矩阵形式为:

$$\begin{bmatrix} 4.0 & 1.0 \\ 1.0 & 4.0 \end{bmatrix} \begin{bmatrix} m_1 \\ m_1 \end{bmatrix} = \begin{bmatrix} 6.0 \\ -12.0 \end{bmatrix}$$

可容易求出 $m_1 = 2.4$ 和 $m_2 = -3.6$。由于 $m_0 = S''(0) = 0$ 和 $m_3 = S''(3) = 0$,所以使用式(16)求解样条系数的结果为:

$$\begin{aligned} S_0(x) &= 0.4x^3 + 0.1x, & 0 \leqslant x \leqslant 1 \\ S_1(x) &= -(x-1)^3 + 1.2(x-1)^2 + 1.3(x-1) + 0.5, & 1 \leqslant x \leqslant 2 \\ S_2(x) &= 0.6(x-2)^3 - 1.8(x-2)^2 + 0.7(x-2) + 2.0, & 2 \leqslant x \leqslant 3 \end{aligned} \tag{19}$$

自然三次样条曲线如图 5.13 所示。

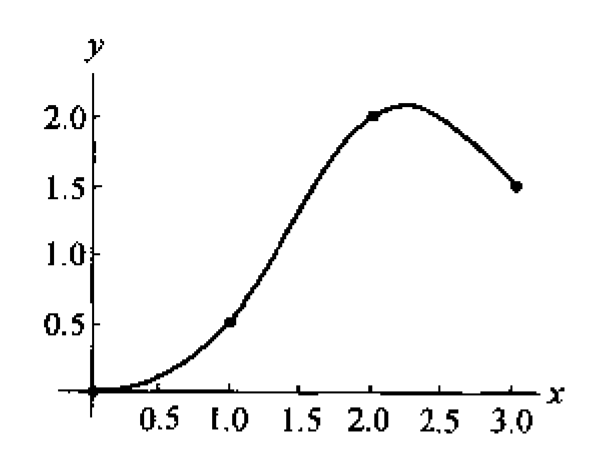


图 5.12 压紧三次样条,边界条件为 $S'(0) = 0.2$ 和 $S'(3) = -1$

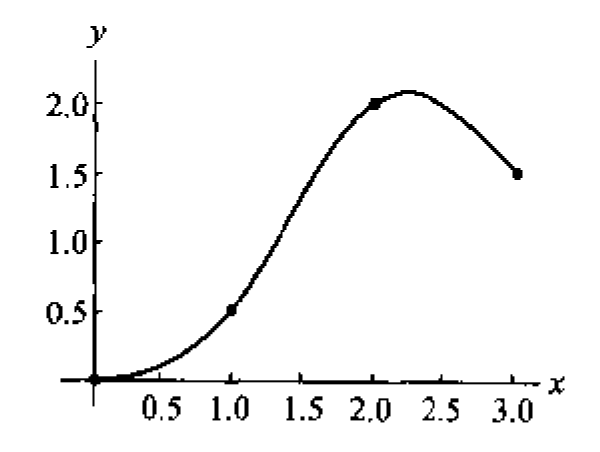


图 5.13 自然三次样条,边界条件为 $S''(0) = 0$ 和 $S''(3) = 0$

例 5.9 求外推三次样条曲线,经过点(0,0.0),(1,0.5),(2,2.0),(3,1.5)。

使用例 5.7 中的值$\{h_k\}$,$\{d_k\}$和$\{u_k\}$并根据引理 5.3,可得线性方程组:

$$(3+2+1)m_1+(1-1)m_2=6.0$$
$$(1-1)m_1+(2+3+1)m_2=-12.0$$

其矩阵形式为：

$$\begin{bmatrix}6.0 & 0.0\\ 0.0 & 6.0\end{bmatrix}\begin{bmatrix}m_1\\ m_2\end{bmatrix}=\begin{bmatrix}6.0\\ -12.0\end{bmatrix}$$

可求出解为 $m_1=1.0$ 和 $m_2=-2.0$。现在将它们代入表5.8的 (iii) 中的方程组，并计算 m_0 和 m_3：

$$m_0=1.0-(-2.0-1.0)=4.0$$
$$m_3=-2.0+(-2.0-1.0)=-5.0$$

最后将 $\{m_k\}$ 代入式(16)求解样条系数可得：

$$\begin{aligned}
S_0(x)&=-0.5x^3+2.0x^2-x, & 0\leqslant x\leqslant 1\\
S_1(x)&=-0.5(x-1)^3+0.5(x-1)^2+1.5(x-1)+0.5, & 1\leqslant x\leqslant 2\\
S_2(x)&=-0.5(x-2)^3-(x-2)^2+(x-2)+2.0, & 2\leqslant x\leqslant 3
\end{aligned}\tag{20}$$

外推三次样条曲线如图5.14所示。

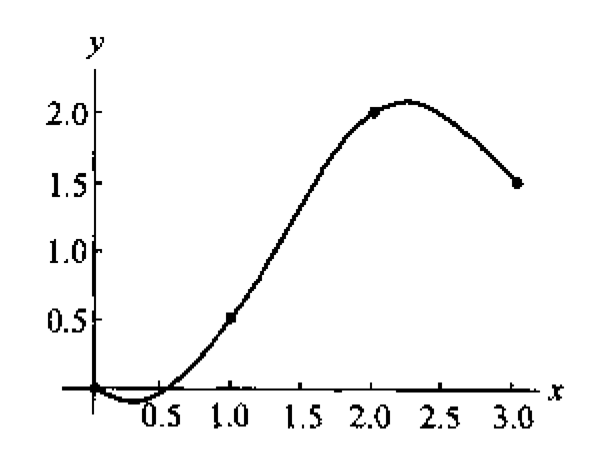


图5.14 外推三次样条曲线

例5.10 求抛物线终结样条曲线，经过点(0,0.0)，(1,0.5)，(2,2.0)，(3,1.5)。

使用例5.7中的值 $\{h_k\}$，$\{d_k\}$ 和 $\{u_k\}$，根据引理5.4可得：

$$(3+2)m_1+m_2=6.0$$
$$m_1+(2+3)m_2=-12.0$$

其矩阵形式为：

$$\begin{bmatrix}5.0 & 1.0\\ 1.0 & 5.0\end{bmatrix}\begin{bmatrix}m_1\\ m_2\end{bmatrix}=\begin{bmatrix}6.0\\ -12.0\end{bmatrix}$$

可得到结果为 $m_1=1.75$ 和 $m_2=-2.75$。由于在每个端点的子区间内 $S''(x)\equiv 0$，根据表5.8 (iv) 中的方程组，可得 $m_0=m_1=1.75$ 和 $m_3=m_2=-2.75$。然后将 $\{m_k\}$ 的值代入式(16)可得：

$$\begin{aligned}
S_0(x)&=0.875x^2-0.375x, & 0\leqslant x\leqslant 1\\
S_1(x)&=-0.75(x-1)^3+0.875(x-1)^2+1.375(x-1)+0.5, & 1\leqslant x\leqslant 2\\
S_2(x)&=-1.375(x-2)^2+0.875(x-2)+2.0, & 2\leqslant x\leqslant 3
\end{aligned}\tag{21}$$

抛物线终结样条曲线如图5.15所示。

例 5.11 求端点曲率调整样条曲线，经过点(0,0.0),(1,0.5),(2,2.0),(3,1.5)，而且二阶导数边界条件 $S''(0)=-0.3$ 和 $S''(3)=3.3$。

使用例 5.7 中的值 $\{h_k\}$，$\{d_k\}$ 和 $\{u_k\}$，并根据引理 5.5 可得：

$$2(1+1)m_1+m_2=6.0-(-0.3)=6.3$$
$$m_1+2(1+1)m_2=-12.0-(3.3)=-15.3$$

其矩阵形式为：

$$\begin{bmatrix}4.0 & 1.0\\ 1.0 & 4.0\end{bmatrix}\begin{bmatrix}m_1\\ m_2\end{bmatrix}=\begin{bmatrix}6.3\\ -15.3\end{bmatrix}$$

方程组的解为 $m_1=2.7$ 和 $m_2=-4.5$。

根据边界条件可得 $m_0=S''(0)=-0.3$ 和 $m_3=S''(3)=3.3$。将 $\{m_k\}$ 的值代入式(16)可得：

$$\begin{aligned}&S_0(x)=0.5x^3-0.15x^2+0.15x, && 0\leqslant x\leqslant 1\\ &S_1(x)=-1.2(x-1)^3+1.35(x-1)^2+1.35(x-1)+0.5, && 1\leqslant x\leqslant 2\\ &S_2(x)=1.3(x-2)^3-2.25(x-2)^2+0.45(x-2)+2.0, && 2\leqslant x\leqslant 3\end{aligned} \tag{22}$$

端点曲率调整样条曲线如图 5.16 所示。

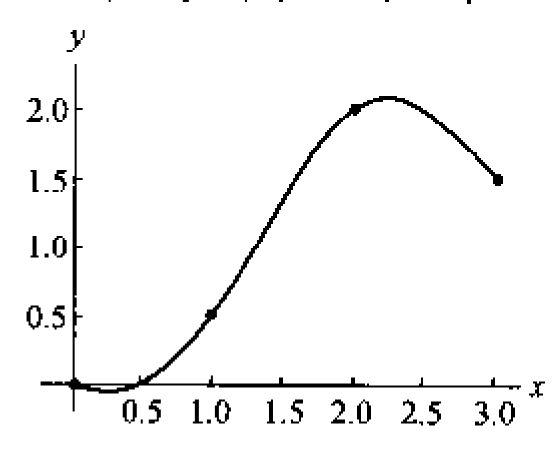


图 5.15 抛物线终结样条曲线

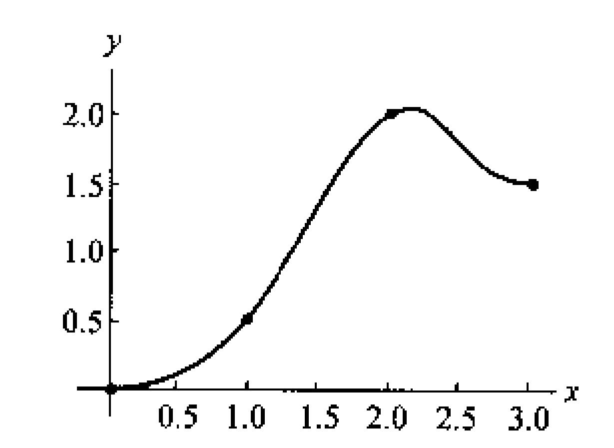


图 5.16 端点曲率调整样条曲线，满足 $S''(0)=-0.3$ 和 $S''(3)=3.3$

5.3.6 三次样条曲线的适宜性

样条曲线的实用性在于它们具有的摆动行为极小。因此，所有的函数 $f(x)$ 在区间 $[a,b]$ 内二次连续可微，同时经过给定的数据点集 $\{x_k, y_k\}_{k=0}^{N}$，而且三次样条曲线的摆动相对较小。下面的结论解释了这一现象。

定理 5.4(三次样条曲线的极小性质) 设 $f\in C^2[a,b]$，且 $S(x)$ 是惟一经过点 $\{(x_k, f(x_k))\}_{k=0}^{N}$ 的函数 $f(x)$ 的三次样条插值曲线，并且满足夹紧端点条件 $S'(a)=f'(a)$ 和 $S'(b)=f'(b)$，则：

$$\int_a^b (S''(x))^2dx\leqslant\int_a^b (f''(x))^2dx \tag{23}$$

证明：利用分部积分法并根据端点条件可得：

$$\int_a^b S''(x)(f''(x)-S''(x))dx$$

$$= S''(x)(f'(x) - S'(x))\Big|_{x=a}^{x=b} - \int_a^b S'''(x)(f'(x) - S'(x))dx$$

$$= 0 - 0 - \int_a^b S'''(x)(f'(x) - S'(x))dx$$

由于在子区间$[x_k, x_{k+1}]$内，$S'''(x) = 6S_{k,3}$，所以可推导出：

$$\int_{x_k}^{x_{k+1}} S'''(x)(f'(x) - S'(x))dx = 6S_{k,3}(f(x) - S(x))\Big|_{x=x_k}^{x=x_{k+1}} = 0$$

其中 $k = 0, 1, \cdots, N-1$。因此$\int_a^b S''(x)(f''(x) - S''(x))dx = 0$，而且可推导出：

$$\int_a^b S''(x)f''(x)dx = \int_a^b (S''(x))^2 dx \tag{24}$$

由于$0 \leqslant (f''(x) - S''(x))^2$，我们可得到积分关系：

$$0 \leqslant \int_a^b (f''(x) - S''(x))^2 dx$$

$$= \int_a^b (f''(x))^2 dx - 2\int_a^b f''(x)S''(x)dx + \int_a^b (S''(x))^2 dx \tag{25}$$

现在将式(24)的结果代入式(25)可得：

$$0 \leqslant \int_a^b (f''(x))^2 dx - \int_a^b (S''(x))^2 dx$$

重写上式则可容易得到关系式(23)，定理得证。

下面的程序构造了一个根据数据点$\{(x_k, y_k)\}_{k=0}^N$的压紧三次样条插值。输出矩阵 S 的第 $k-1$ 行是 $S_k(x)$，其中 $k = 0, 1, \cdots, N-1$ 的按降序排列的系数。在练习中，读者要根据其他如表 5.8 中和引理 5.2 到引理 5.5 中描述的端点条件，修改下面的程序。

程序 5.3(压紧三次样条曲线) 根据 $N+1$ 个点$\{(x_k, y_k)_{k=0}^N\}$，构造并计算夹紧三次样条插值

```
function S = csfit(X,Y,dx0,dxn)
% Input   - X is the 1xn abscissa vector
%         - Y is the 1xn ordinate vector
%         - dx0 = S' (x0) first derivative boundary condition
%         - dxn = S' (xn) first derivative boundary condition
% Output  - S:rows of S are the coefficients,in descending
%           order,for the cubic interpolants
N = length(X) - 1;
H = diff(X);
D = diff(Y)./H;
A = H(2:N-1);
B = 2 * (H(1:N-1) + H(2:N));
C = H(2:N);
U = 6 * diff(D);
% Clamped spline endpoint constraints
B(1) = B(1) - H(1)/2;
U(1) = U(1) - 3 * (D(1) - dx0);
B(N-1) = B(N-1) - H(N)/2;
```

```
U(N-1)=U(N-1)-3*(dxn-D(N));
for k=2:N-1
    temp=A(k-1)/B(k-1);
    B(k)=B(k)-temp*C(k-1);
    U(k)=U(k)-temp*I(k-1);
end
M(N)=U(N-1)/B(N-1);
for k=N-2:-1:1
    M(k+1)=(U(k)-C(k)*M(k+2))/B(k);
end
M(1)=3*(D(1)-dxo)/H(1)-M(2)/2;
M(N+1)=3*(dxn-D(N))/H(N)-M(N)/2;
for k=0:N-1
    S(k+1,1)=(M(k+2)-M(k+1))/(6*H(k+1));
    S(k+1,2)=M(k+1)/2;
    S(k+1,3)=D(k+1)-H(k+1)*(2*M(k+1)+M(k+2))/6;
    S(k+1,4)=Y(k+1);
end
```

例 5.12 求解压紧三次样条曲线,经过点(0,0.0),(1,0.5),(2,2.0),(3,1.5),而且一阶导数边界条件 $S'(0)=0.2$ 和 $S'(3)=-1$。

在 MATLAB 中进行计算:

```
>>X=[0  1  2  3];Y=[0  0.5  2.0  1.5];dx0=0.2;dxn=-1;
>>S=csfit(X,Y,dxo,dxn)
S=
  0.4800  -0.1800 0.2000 0
  -1.0400 1.2600 1.2800 0.5000
  0.6800  -1.8600 0.6800 2.0000
```

要注意 S 中行的元素是例 5.7 中式(18)对应方程的系数。下面的命令显示了如何使用polyval命令画出三次样条插值。产生的图与图 5.12 相同。命令如下:

```
>>x1=0:.01:1;y1=polyval(S(1,:),x1-X(1));
>>x2=1:.01:2;y2=polyval(S(2,:),x2-X(2));
>>x3=2:.01:3;y3=polyval(S(3,:),x3-X(3));
>>plot(x1,y1,x2,y2,;x3,y3,x,y,',')
```

5.3.7 样条函数插值的练习

1. 设有多项式 $S(x)=a_0+a_1x+a_2x^2+a_3x^3$。

   (a) 证明根据条件 $S(1)=1, S'(1)=0, S(2)=2, S'(2)=0$ 可得到如下方程组:

$$a_0+a_1+a_2+a_3=1$$
$$a_1+2a_2+3a_3=0$$
$$a_0+2a_1+4a_2+8a_3=2$$
$$a_1+4a_2+12a_3=0$$

   (b) 求解(a)中的方程组,并根据结果画出三次多项式曲线。

2. 设有多项式 $S(x)=a_0+a_1x+a_2x^2+a_3x^3$。

(a) 证明根据条件 $S(1)=3, S'(1)=-4, S(2)=1, S'(2)=0$ 可得到如下方程组：

$$\begin{aligned} a_0 + a_1 + a_2 + a_3 &= 3 \\ a_1 + 2a_2 + 3a_3 &= -4 \\ a_0 + 2a_1 + 4a_2 + 8a_3 &= 1 \\ a_1 + 4a_2 + 12a_3 &= 2 \end{aligned}$$

(b) 求解(a)中的方程组,并根据结果画出三次多项式曲线。

3. 判断下列哪些函数是三次样条。提示：函数 $f(x)$ 是否满足定义 5.1 中的 5 个性质？

(a) $f(x)=\begin{cases} \dfrac{19}{2}-\dfrac{81}{4}x+15x^2-\dfrac{13}{4}x^3 & 1\leqslant x\leqslant 2 \\ \dfrac{-77}{2}+\dfrac{207}{4}x+21x^2+\dfrac{11}{4}x^3 & 2\leqslant x\leqslant 3 \end{cases}$

(b) $f(x)=\begin{cases} 11-24x+18x^2-4x^3 & 1\leqslant x\leqslant 2 \\ -54+72x-30x^2+4x^3 & 2\leqslant x\leqslant 3 \end{cases}$

(c) $f(x)=\begin{cases} 18-\dfrac{75}{2}x+26x^2-\dfrac{11}{2}x^3 & 1\leqslant x\leqslant 2 \\ -70+\dfrac{189}{2}x-40x^2+\dfrac{11}{2}x^3 & 2\leqslant x\leqslant 3 \end{cases}$

(d) $f(x)=\begin{cases} 13-31x+23x^2-5x^3 & 1\leqslant x\leqslant 2 \\ -35+51x-22x^2+3x^3 & 2\leqslant x\leqslant 3 \end{cases}$

4. 求压紧三次样条曲线,经过点$(-3,2),(-2,0),(1,3),(4,1)$,而且一阶导数边界条件 $S'(-3)=-1$ 和 $S'(4)=1$。

5. 求自然三次样条曲线,经过点$(-3,2),(-2,0),(1,3),(4,1)$,而且自由边界条件 $S''(-3)=0$ 和 $S''(4)=0$。

6. 求外推三次样条曲线,经过点$(-3,2),(-2,0),(1,3),(4,1)$。

7. 求抛物线终结三次样条曲线,经过点$(-3,2),(-2,0),(1,3),(4,1)$。

8. 求曲率调整三次样条曲线,经过点$(-3,2),(-2,0),(1,3),(4,1)$,而且二阶导数边界条件 $S''(-3)=-1$ 和 $S''(4)=2$。

9. (a) 求压紧三次样条曲线,经过点集 $\{(x_k,f(x_k))\}_{k=0}^{3}$,其中 $f(x)=x+\dfrac{2}{x}$,横坐标为 $x_0=1/2, x_1=1, x_2=3/2, x_3=2$。一阶导数边界条件为 $S'(x_0)=f'(x_0)$ 和 $S'(x_3)=f'(x_3)$。在同一坐标系下,画出函数 f 和压紧三次样条插值。

(b) 求自然三次样条曲线,经过点集 $\{(x_k,f(x_k))\}_{k=0}^{3}$,其中 $f(x)=x+\dfrac{2}{x}$,横坐标为 $x_0=1/2, x_1=1, x_2=3/2, x_3=2$。二阶导数边界条件为 $S''(x_0)=0$ 和 $S''(x_3)=0$。在同一坐标系下,画出函数 f 和自然三次样条插值。

10. (a) 求压紧三次样条曲线,经过点集 $\{(x_k,f(x_k))\}_{k=0}^{3}$,其中 $f(x)=\cos(x^2)$,横坐标为 $x_0=0, x_1=\sqrt{\pi/2}, x_2=\sqrt{3\pi/2}, x_3=\sqrt{5\pi/2}$。一阶导数边界条件为 $S'(x_0)=f'(x_0)$和

$S'(x_3)=f'(x_3)$。在同一坐标系下,画出函数 f 和压紧三次样条插值。

(b) 求自然三次样条曲线,经过点集$\{(x_k,f(x_k))\}_{k=0}^{3}$,其中 $f(x)=\cos(x^2)$,横坐标为 $x_0=0$, $x_1=\sqrt{\pi/2}$, $x_2=\sqrt{3\pi/2}$, $x_3=\sqrt{5\pi/2}$。二阶导数边界条件为 $S''(x_0)=0$ 和 $S''(x_3)=0$。在同一坐标系下,画出函数 f 和自然三次样条插值。

11. 利用下列替换表达式:

$$x_{k+1}-x=h_k+(x_k-x)$$

和:

$$(x_{k+1}-x)^3=h_k^3+3h_k^2(x_k-x)+3h_k(x_k-x)^2+(x_k-x)^3$$

证明当式(8)扩展为(x_k-x)的幂的形式时,它的系数是式(16)中给出的系数。

12. 设三次函数 $S_k(x)$在区间$[x_k,x_{k+1}]$内:

(a) 给出计算 $\int_{x_k}^{x_{k+1}} S_k(x)\,dx$ 的一个公式。

然后根据下面给出的练习的(a)部分计算 $\int_{x_0}^{x_3} S(x)\,dx$。

(b) 练习 10　　(c) 练习 11

13. 如何结合表 5.8 的策略(i)和式(12)得到引理 5.1 中的方程。

14. 如何结合表 5.8 的策略(iii)和式(12)得到引理 5.1 中的方程。

15. (a) 使用点 $x_0=-2$ 和 $x_1=0$,证明函数 $f(x)=x^3-x$ 在区间$[-2,0]$内的压紧三次样条插值是其自身。

(b) 使用点 $x_0=-2, x_1=0, x_2=2$,证明函数 $f(x)=x^3-x$ 在区间$[-2,2]$内的压紧三次样条插值是其自身。注意:f 在 x_1处有一个拐点(inflection point)。

(c) 根据(a)和(b)的结论,证明任意三阶多项式 $f(x)=a_0+a_1x+a_2x^2+a_3x^3$ 在任意闭区间$[a,b]$内的压紧三次样条插值是其自身。

(d) 从引理 5.2 到引理 5.5 描述的其他 4 种三次样条插值有类似(c)的结论吗? 请叙述。

5.3.8 算法和程序

1. 一个轿车在时间 t_k 时经过的距离为 d_k,如下表所示。使用程序 5.3,并根据一阶导数边界条件 $S'(0)=0$ 和 $S'(8)=98$,求这些数据的压紧三次样条插值:

时间 t_k	0	2	4	6	8
距离 d_k	0	40	160	300	480

2. 修改程序 5.3,根据给定数据点集,求 (a) 自然三次样条插值,(b)外推三次样条插值,(c) 抛物线终结三次样条插值,(d)端点曲率调整三次样条插值。

3. 使用问题 2 中的程序,根据点(0,1),(1,0),(2,0),(3,1),(4,2),(5,2),(6,1),求 5 种不同的三次样条插值,其中 $S'(0)=-0.6$, $S'(6)=-1.8$, $S''(0)=1$ 和 $S''(6)=-1$。在同一坐标系中画出这 5 个三次样条插值和这些数据点。

4. 使用问题 2 中的程序,根据点(0,0),(1,4),(2,8),(3,9),(4,9),(5,8),(6,6),求 5 种

不同的三次样条插值，其中 $S'(0)=1$，$S'(6)=-2$，$S''(0)=1$ 和 $S''(6)=-1$。在同一坐标系中画出这5个三次样条插值和这些数据点。

5. 下面的表给出了在洛杉矶的郊区12小时内每个小时的温度（华氏温度）。根据这些数据求自然三次样条插值。在同一坐标系，画出自然三次样条插值和这些数据。根据自然三次样条插值和练习(12)的(a)部分的结论求12小时内的平均温度近似值。给定温度表为：

时 间	温 度	时 间	温 度
1	58	7	57
2	58	8	58
3	58	9	60
4	58	10	64
5	57	11	67
6	57	中午	68

6. 使用压紧三次样条插值近似在区间$[-3,3]$内的函数 $f(x)=x-\cos(x^3)$。

5.4 傅里叶级数和三角多项式

科学家和工程师经常研究一些具有周期性的现象，如光和声音。它们可用函数 $f(x)$ 描述，$f(x)$ 具有周期性：

对所有的 x 有
$$g(x+P)=g(x) \tag{1}$$
数 P 称为函数的周期。

设函数的周期为 2π。如果 $g(x)$ 的周期为 P，则 $f(x)=g(Px/2\pi)$ 的周期为 2π。可通过下式进行验证：
$$f(x+2\pi)=g\left(\frac{Px}{2\pi}+P\right)=g\left(\frac{Px}{2\pi}\right)=f(x) \tag{2}$$
自此以后，在这一节中，假设函数 $f(x)$ 的周期是 2π，即：

对所有的 x 有
$$f(x+2\pi)=f(x) \tag{3}$$
通过重复函数在某个长度为 2π 的区间内的图形可构成整个函数的图形，如图5.17所示。

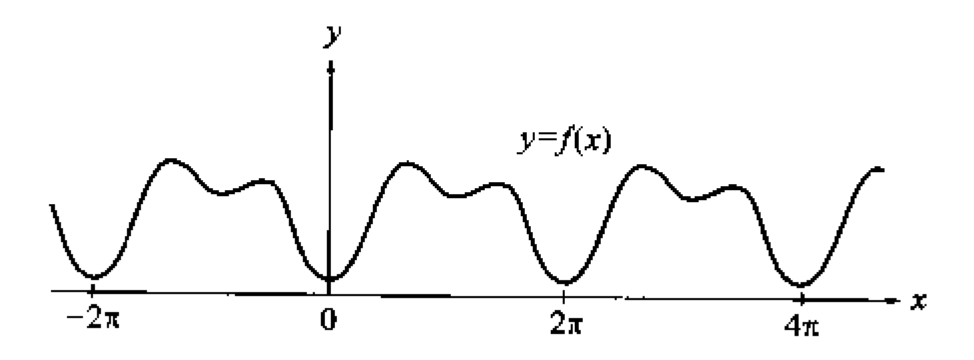


图5.17 周期为 2π 的连续函数 $f(x)$

例如函数 $\sin(jx)$ 和 $\cos(jx)$，其中 j 是整数，是周期为 2π 的函数。这时会产生下面的问题：一个周期函数是否能表示为包含 $a_j\cos(jx)$ 和 $b_j\sin(jx)$ 的项的和？下面将看到答案是肯定的。

定义 5.2(分段连续) 如果存在值 $t_0, t_1, \cdots, t_K$ 满足 $a = t_0 < t_1 < \cdots < t_K = b$，函数$f(x)$在每个开区间 $t_{i-1} < x < t_i, i = 1,2,\cdots,K$ 内是连续的，而且函数在每个端点 t_i 有左极限和右极限，则称函数 $f(x)$在区间$[a,b]$内分段连续。如图 5.18 所示。

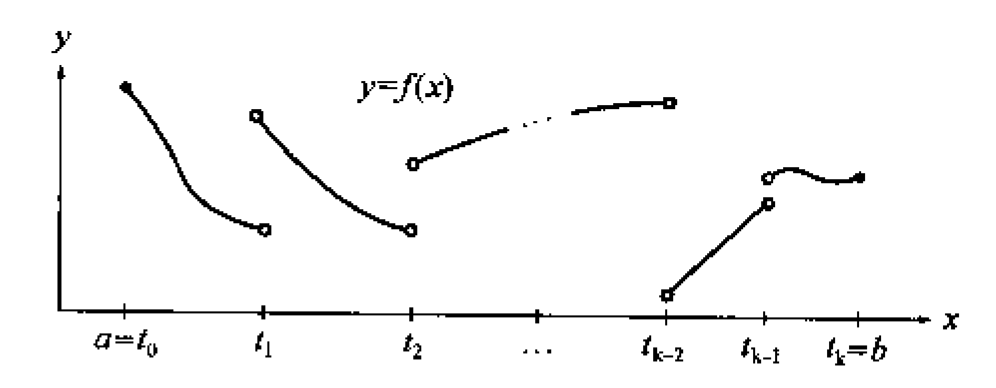


图 5.18 在区间$[a,b]$内的分段连续函数

定义 5.3(傅里叶级数) 设 $f(x)$是周期函数，周期为 2π，而且 $f(x)$在区间$[-\pi,\pi]$内分段连续。则 $f(x)$的傅里叶级数 $S(x)$表示为：

$$S(x) = \frac{a_0}{2} + \sum_{j=1}^{\infty}(a_j \cos(jx) + b_j \sin(jx)) \tag{4}$$

这里的系数 a_j和 b_j可用欧拉公式计算得到：

$$a_j = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)\cos(jx)\,dx, \qquad j = 0,1,\cdots \tag{5}$$

和

$$b_j = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)\sin(jx)\,dx, \qquad j = 1,2,\cdots \tag{6}$$

引入傅里叶级数式(4)中的常数项 $a_0/2$ 的因子$\frac{1}{2}$，使得可通过设 $j=0$，并计算式(5)得到 a_0。下面讨论了傅里叶级数的收敛性。

定理 5.5 (傅里叶级数扩展) 设 $S(x)$是 $f(x)$在区间$[-\pi,\pi]$内的傅里叶级数。如果$f'(x)$在区间$[-\pi,\pi]$内是分段连续的，而且在区间内的每个端点有左导数和右导数，则$S(x)$对于所有 $x\in[-\pi,\pi]$是收敛的。对于所有的点 $x\in[-\pi,\pi]$存在关系式：

$$S(x) = f(x)$$

其中 $f(x)$是连续的。如果 $x = a$ 是函数f 的不连续点，则：

$$S(a) = \frac{f(a^-) + f(a^+)}{2}$$

这里 $f(a^-)$和 $f(a^+)$分别表示左极限和右极限。这样，可得到傅里叶级数扩展表达式：

$$f(x) = \frac{a_0}{2} + \sum_{j=1}^{\infty}(a_j \cos(jx) + b_j \sin(jx)) \tag{7}$$

在本小节最后，对式(5)和式(6)的推导进行了简明描述。

例 5.13 设在区间 $-\pi < x < \pi$ 内有函数$f(x) = x/2$，周期性满足 $f(x+2\pi) = f(x)$，证明它的傅里叶级数可表示为：

$$f(x) = \sum_{j=1}^{\infty} \frac{(-1)^{j+1}}{2}\sin(jx) = \sin(x) - \frac{\sin(2x)}{2} + \frac{\sin(3x)}{3} - \cdots$$

利用欧拉公式和分部积分法可得：

$$a_j = \frac{1}{\pi}\int_{-\pi}^{\pi} \frac{x}{2}\cos(jx)\,dx = \frac{x\sin(jx)}{2\pi j} + \frac{\cos(jx)}{2\pi j^2}\Bigg|_{-\pi}^{\pi} = 0$$

其中 $j=1,2,3,\cdots$，并且：

$$b_j = \frac{1}{\pi}\int_{-\pi}^{\pi} \frac{x}{2}\sin(jx)\,dx = \frac{-x\cos(jx)}{2\pi j} + \frac{\sin(jx)}{2\pi j^2}\Bigg|_{-\pi}^{\pi} = \frac{(-1)^{j+1}}{j}$$

其中 $j=1,2,3,\cdots$。系数 a_0 为：

$$a_0 = \frac{1}{\pi}\int_{-\pi}^{\pi} \frac{x}{2}\,dx = \frac{x^2}{4\pi}\Bigg|_{-\pi}^{\pi} = 0$$

通过计算可知余弦函数的系数为零。函数 $f(x)$ 的部分和：

$$S_2(x) = \sin(x) - \frac{\sin(2x)}{2}$$

$$S_3(x) = \sin(x) - \frac{\sin(2x)}{2} + \frac{\sin(3x)}{3}$$

$$S_4(x) = \sin(x) - \frac{\sin(2x)}{2} + \frac{\sin(3x)}{3} - \frac{\sin(4x)}{4}$$

如图 5.19 所示。

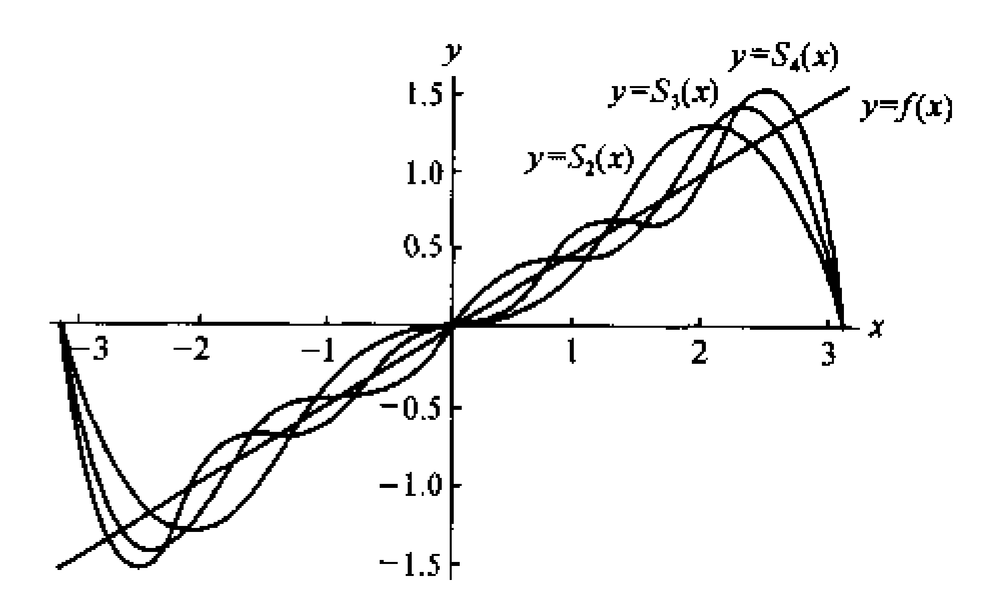


图 5.19　在区间 $[-\pi,\pi]$ 内的函数 $f(x)=x/2$ 和三角近似 $S_2(x), S_3(x), S_4(x)$

下面是傅里叶级数的一些一般性质。相关证明留作练习。

定理 5.6（余弦级数）　设 $f(x)$ 是偶函数，即对所有的 x 有 $f(-x)=f(x)$。如果 $f(x)$ 的周期为 2π，而且 $f(x)$ 和 $f'(x)$ 是分段连续，则 $f(x)$ 的傅里叶级数只包含余弦项：

$$f(x) = \frac{a_0}{2} + \sum_{j=1}^{\infty} a_j\cos(jx) \tag{8}$$

其中：

$$a_j = \frac{2}{\pi}\int_0^{\pi} f(x)\cos(jx)\,dx, \quad j = 0,1,\cdots$$

定理 5.7（正弦级数）　设 $f(x)$ 是奇函数，即对所有的 x 有 $f(-x)=-f(x)$。如果 $f(x)$ 的周期为 2π，而且 $f(x)$ 和 $f'(x)$ 是分段连续的，则 $f(x)$ 的傅里叶级数只包含正弦项：

$$f(x) = \sum_{j=1}^{\infty} b_j \sin(jx) \tag{10}$$

其中：

$$b_j = \frac{2}{\pi}\int_0^{\pi} f(x)\sin(jx)\,dx, j = 1,2,\cdots \tag{11}$$

例 5.14 设在区间 $-\pi < x < \pi$ 内有函数 $f(x) = |x|$，周期性满足 $f(x + 2\pi) = f(x)$，证明它具有傅里叶余弦级数表达式：

$$\begin{aligned} f(x) &= \frac{\pi}{2} - \frac{4}{\pi}\sum_{j=1}^{\infty} \frac{\cos((2j-1)x)}{(2j-1)^2} \\ &= \frac{\pi}{2} - \frac{4}{\pi}\left(\cos(x) + \frac{\cos(3x)}{3^2} + \frac{\cos(5x)}{5^2} + \cdots\right) \end{aligned} \tag{12}$$

由于函数 $f(x)$ 为偶函数，因此根据定理 5.6 只需计算系数 $\{a_j\}$：

$$\begin{aligned} a_j &= \frac{2}{\pi}\int_0^{\pi} x\cos(jx)\,dx = \frac{2x\sin(jx)}{\pi j} + \frac{2\cos(jx)}{\pi j^2}\bigg|_0^{\pi} \\ &= \frac{2\cos(j\pi) - 2}{\pi j^2} = \frac{2((-1)^j - 1)}{\pi j^2}, \; j = 1,2,3,\cdots \end{aligned}$$

由于当 j 是偶数时有 $((-1)^j - 1) = 0$，所以余弦级数只有奇数项。而奇数系数具有下列模式：

$$a_1 = \frac{-4}{\pi}, \quad a_3 = \frac{-4}{\pi 3^2}, \quad a_5 = \frac{-4}{\pi 5^2}, \cdots$$

系数 a_0 为：

$$a_0 = \frac{2}{\pi}\int_0^{\pi} x\,dx = \frac{x^2}{\pi}\bigg|_0^{\pi} = \pi$$

这样，可得到式(12)中的系数。

定理 5.5 中欧拉公式的证明：设傅里叶级数存在且收敛。为确定 a_0，可对式(7)的两边进行积分得到：

$$\begin{aligned} \int_{-\pi}^{\pi} f(x)\,d_x &= \int_{-\pi}^{\pi}\left(\frac{a_0}{2} + \sum_{j=1}^{\infty}(a_j\cos(jx) + b_j\sin(jx))\right)dx \\ &= \int_{-\pi}^{\pi}\frac{a_0}{2}dx + \sum_{j=1}^{\infty} a_j\int_{-\pi}^{\pi}\cos(jx)\,dx + \sum_{j=1}^{\infty} b_j\int_{-\pi}^{\pi}\sin(jx)\,dx \\ &= \pi a_0 + 0 + 0 \end{aligned} \tag{13}$$

通过对上式进行一致收敛（参见相关的高等教材）来调整求和与积分的顺序，可得到：

$$a_0 = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)\,dx \tag{14}$$

为得到 a_m，设 $m > 0$ 是一个固定整数，在式(7)的两边乘以 $\cos(mx)$ 并积分可得：

$$\int_{-\pi}^{\pi} f(x)\cos(mx)\,dx = \frac{a_0}{2}\int_{-\pi}^{\pi}\cos(mx)\,dx + \sum_{j=1}^{\infty} a_j\int_{-\pi}^{\pi}\cos(jx)\cos(mx)\,dx$$

$$+\sum_{j=1}^{\infty} b_j \int_{-\pi}^{\pi} \sin(jx)\cos(mx)\,dx \tag{15}$$

根据三角函数的正交性质，可对式(15)进行化简。式(15)右边的第一项的值为：

$$\frac{a_0}{2}\int_{-\pi}^{\pi}\cos(mx)\,dx = \left.\frac{a_0\sin(mx)}{2m}\right|_{-\pi}^{\pi} = 0 \tag{16}$$

通过利用下列三角恒等式可得到包含 $\cos(jx)\cos(mx)$ 的项的值：

$$\cos(jx)\cos(mx) = \frac{1}{2}\cos((j+m)x) + \frac{1}{2}\cos((j-m)x) \tag{17}$$

当 $j \neq m$ 时，根据式(17)可得：

$$\begin{aligned} a_j\int_{-\pi}^{\pi}\cos(jx)\cos(mx)\,dx &= \frac{1}{2}a_j\int_{-\pi}^{\pi}\cos((j+m)x)\,dx \\ &\quad + \frac{1}{2}a_j\int_{-\pi}^{\pi}\cos((j-m)x)\,dx = 0 + 0 = 0 \end{aligned} \tag{18}$$

当 $j = m$ 时，积分的值为：

$$a_m\int_{-\pi}^{\pi}\cos(jx)\cos(mx)\,dx = a_m\pi \tag{19}$$

通过利用下列三角恒等式可得式(15)中右边包含 $\sin(jx)\cos(mx)$ 的项的值：

$$\sin(jx)\cos(mx) = \frac{1}{2}\sin((j+m)x) + \frac{1}{2}\sin((j-m)x) \tag{20}$$

对于式(20)中所有的 j 和 m 有：

$$\begin{aligned} b_j\int_{-\pi}^{\pi}\sin(jx)\cos(mx)\,dx &= \frac{1}{2}b_j\int_{-\pi}^{\pi}\sin((j+m)x)\,dx \\ &\quad + \frac{1}{2}b_j\int_{-\pi}^{\pi}\sin((j-m)x)\,dx = 0 + 0 = 0 \end{aligned} \tag{21}$$

因此，根据式(16)、(18)、(19)和(21)，可得到：

$$\pi a_m = \int_{-\pi}^{\pi} f(x)\cos(mx)\,dx,\ m = 1,2,\cdots \tag{22}$$

所以欧拉公式(5)成立。同理可证欧拉公式(6)。

5.4.1 三角多项式逼近

定义 5.4（三角多项式） 具有如下形式的级数：

$$T_M(x) = \frac{a_0}{2} + \sum_{j=1}^{M}(a_j\cos(jx) + b_j\sin(jx)) \tag{23}$$

称为 M 阶三角多项式。

定理 5.8（离散傅里叶级数） 设有 $N+1$ 个点 $\{(x_j, y_j)\}_{j=0}^{N}$，其中 $y_j = f(x_j)$，而且横坐标之间等距，即：

$$x_j = -\pi + \frac{2j\pi}{N},\ j = 0,1,\cdots,N \tag{24}$$

如果 $f(x)$ 的周期为 2π，而且 $2M < N$，则存在式(23)的三角多项式 $T_M(x)$ 使得下式值最小：

$$\sum_{k=1}^{N}(f(x_k)-T_M(x_k))^2 \tag{25}$$

多项式的系数 a_j 和 b_j 可通过如下公式计算：

$$a_j=\frac{2}{N}\sum_{k=1}^{N}f(x_k)\cos(jx_k),\quad j=0,1,\cdots,M \tag{26}$$

和：

$$b_j=\frac{2}{N}\sum_{k=1}^{N}f(x_k)\sin(jx_k),\quad j=1,2,\cdots,M \tag{27}$$

尽管式(26)和式(27)用最小二乘法定义，它们可被看作是欧拉公式(5)和(6)的积分的数值近似值。欧拉公式给出了连续函数的傅里叶级数的系数，而式(26)和式(27)给出了对数据点集进行曲线拟合的三角多项式系数。下面的例子根据函数 $f(x)+x/2$ 生成数据点集。当使用更多的数据点时，三角多项式的系数更接近傅里叶级数的系数。

例 5.15 根据 12 个等距横坐标点 $x_k=-\pi+k\pi/6,k=1,2,\cdots,12$，求解点集 $\{(x_k,f(x_k))\}_{k=1}^{12}$（其中 $f(x)=x/2$）的 5 阶三角多项式逼近。并比较使用 60 个点和 360 个点的结果情况与使用例 5.13 中 $f(x)$ 的傅立叶级数扩展的前 5 项的结果情况。

由于周期扩展已知，在非连续点，函数值 $f(\pi)$ 必须利用下列公式进行计算：

$$f(\pi)=\frac{f(\pi^-)+f(\pi^+)}{2}=\frac{\pi/2-\pi/2}{2}=0 \tag{28}$$

函数 $f(x)$ 是一个奇函数，因此余弦项的系数为零（即对于所有的 j 有 $a_j=0$）。5 阶三角多项式只包含正弦项，结合式(28)和式(27)可得：

$$\begin{aligned}T_5(x)&=0.9770486\sin(x)-04534498\sin(2x)+0.26179938\sin(3x)\\&\quad-0.1511499\sin(4x)+0.0701489\sin(5x)\end{aligned} \tag{29}$$

$T_5(x)$ 的图形如图 5.20 所示。

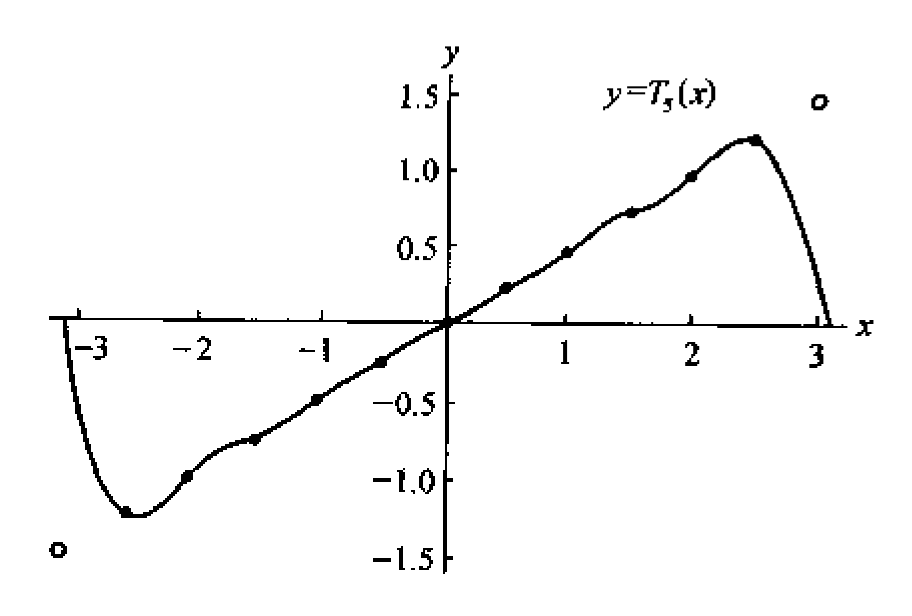


图 5.20 根据位于 $y=x/2$ 上的 12 个数据点创建的 5 阶三角多项式 $T_5(x)$

当插入点增至 60 个和 360 个时，5 阶三角多项式的系数变化很小。当数据点的个数增加越多，三角多项式的系数越接近 $f(x)$ 的傅里叶级数的系数。结果的比较如表 5.9 所示。

表 5.9　比较在区间$[-\pi,\pi]$内的函数 $f(x)=x/2$ 的三角多项式逼近的系数

	三角多项式的系数			傅里叶级数的系数
	12 个点	60 个点	360 个点	
b_1	0.97704862	0.99908598	0.99997462	1.0
b_2	−0.45344984	−0.49817096	−0.49994923	−0.5
b_3	0.26179939	0.33058726	0.33325718	0.33333333
b_4	−0.15114995	−0.24633386	−0.24989845	−0.25
b_5	0.07014893	0.19540972	0.19987306	0.2

下面的程序构造了分别包含 M 阶三角多项式(23)的系数 a_j和 b_j的矩阵 $\boldsymbol{A}$ 和矩阵 $\boldsymbol{B}$。

程序 5.4（三角多项式） 设 $2M+1\leqslant N$，根据 N 个等距的横坐标值 $x_k=-\pi+2\pi k/N$，$k=1,2,\cdots,N$，构造 M 阶三角多项式，其形式为：

$$P(x)=\frac{a_0}{2}+\sum_{j=1}^{M}(a_j\cos(jx)+b_j\sin(jx))$$

```
function [A,B] = tpcoeff(X,Y,M)
% Input   - X is a vector of equally spaced abscissas in [-pi,pi]
%         - Y is a vector of ordinates
%         - M is the degree of the trigonometric polynomial
% Output  - A is a vector containing the coefficients of cos(jx)
%         - B is a vector containing the coefficients of sin(jx)
N = length(X) - 1;
max1 = fix((N-1)/2);
if M > max1
   M = max1;
end
A = zeros(1,M+1);
B = zeros(1,M+1);
Yends = (Y(1) + Y(N+1))/2;
Y(1) = Yends;
Y(N+1) = Yends;
A(1) = sum(Y);
for j = 1:M
   A(j+1) = cos(j*X)*Y';
   B(j+1) = sin(j*X)*Y';
end
   A = 2*A/N;
B = 2*B/N;
A(1) = A(1)/2;
```

下面的短程序计算了程序 5.4 的 M 阶三角多项式 $P(x)$在点 x 处的值。

```
function z = tp(A,B,x,M)
z = A(1);
for j = 1:M
   z = z + A(j+1)*cos(j*x) + B(j+1)*sin(j*x);
end
```

例如，在 MATLAB 命令窗口中输入下面的命令可得到与图 5.20 类似的图形。

```
>>x=-pi:.01:pi;
>>y=tp(A,B,x,M);
>>plot(x,y,X,Y,'o')
```

5.4.2 傅里叶级数和三角多项式的练习

练习 1 到练习 5 中,求给定函数的傅里叶级数表示。提示:参照例 5.13 和例 5.14 描述的过程。在同一坐标系画出每个函数,及傅里叶级数的部分和 $S_2(x),S_3(x),S_4(x)$(如图 5.19 所示)。

1. $f(x)=\begin{cases}-1, & -\pi<x<0\\ 1, & 0<x<\pi\end{cases}$

2. $f(x)=\begin{cases}\dfrac{\pi}{2}+x, & -\pi\leqslant x<0\\ \dfrac{\pi}{2}-x, & 0\leqslant x<\pi\end{cases}$

3. $f(x)=\begin{cases}0, & -\pi\leqslant x<0\\ x, & 0\leqslant x<\pi\end{cases}$

4. $f(x)=\begin{cases}-1, & \dfrac{\pi}{2}<x<\pi\\ 1, & \dfrac{-\pi}{2}<x<\dfrac{\pi}{2}\\ -1, & -\pi<x<\dfrac{-\pi}{2}\end{cases}$

5. $f(x)=\begin{cases}-\pi-x, & -\pi\leqslant x<\dfrac{-\pi}{2}\\ x, & \dfrac{-\pi}{2}\leqslant x<\dfrac{\pi}{2}\\ \pi-x, & \dfrac{\pi}{2}\leqslant x<\pi\end{cases}$

6. 在练习 1 中,设 $x=\pi/2$,证明:

$$\frac{\pi}{4}=1-\frac{1}{3}+\frac{1}{5}-\frac{1}{7}+\cdots$$

7. 在练习 2 中,设 $x=0$,证明:

$$\frac{\pi^2}{8}=1+\frac{1}{3^2}+\frac{1}{5^2}+\frac{1}{7^2}+\cdots$$

8. 求解在一个周期范围的区间 $-\pi\leqslant x<\pi$ 内的周期函数 $f(x)=x^2/4$ 的傅里叶余弦级数。

9. 设 $f(x)$ 是周期函数,周期为 $2P$,即对所有的 x,$f(x+2P)=f(x)$。通过适当的变换,证明函数 f 的欧拉公式(5)和(6)为:

$$a_0=\frac{1}{P}\int_{-P}^{P}f(x)\,dx$$

$$a_j=\frac{1}{P}\int_{-P}^{P}f(x)\cos\left(\frac{j\pi x}{P}\right)dx,\quad j=1,2,\cdots$$

$$b_j=\frac{1}{P}\int_{-P}^{P}f(x)\sin\left(\frac{j\pi x}{P}\right)dx,\quad j=1,2,\cdots$$

在练习 10 到练习 12 中,根据练习 9 的结果,求解给定函数的傅里叶级数。在同一坐标系下画出 $f(x),S_4(x),S_6(x)$。

10. $f(x)=\begin{cases}0, & -2\leqslant x<0\\ 1, & 0\leqslant x<2\end{cases}$　　11. $f(x)=\begin{cases}-1, & -3\leqslant x<-1\\ |x|, & -1\leqslant x<1\\ 1, & 1\leqslant x<3\end{cases}$

12. $f(x)=-x^2+9,\quad -3\leqslant x<3$

13. 证明定理5.6。

14. 证明定理5.7。

5.4.3 算法和程序

1. 使用程序5.4,并且有12个横坐标点,参照例5.15求解根据等距点$\{(x_k,f(x_k))\}_{k=1}^{12}$的5阶三角多项式。其中函数$f(x)$为:(a)练习1,(b)练习2,(c)练习3,(d)练习4。对于每种情况,在同一坐标系下画出$f(x)$,$T_5(x)$和$\{(x_k,f(x_k))\}_{k=1}^{12}$。
2. 第一次使用60个等距数据点,第二次使用360个等距数据点,用程序5.4求解例5.15中$T_5(x)$的系数。
3. 修改程序5.4,使得当等距数据点位于区间$[a,b]$时,可以用它求解周期为$2P=b-a$的三角多项式。
4. 使用修改后的程序5.4求解$T_5(x)$。其中:

   (a) $f(x)$定义在练习10中,有12个等距的数据点。

   (b) $f(x)$定义在练习12中,有60个等距的数据点。

   对每个情况,在同一坐标系下,画出$T_5(x)$和数据点集。
5. 在洛杉矶郊区11月8日的温度(华氏温度)如表5.10所示,采用24小时制。

   (a) 求三角多项式$T_7(x)$。

   (b) 在同一坐标系下,画出图$T_7(x)$和24个数据点。

   (c) 使用本地的温度情况重新求解问题(a)和(b)。
6. 阿拉斯加州的费尔班克斯地区的年度温度(华氏温度)如表5.11所示。一共有13个等距点,即每隔28天采集一次。

   (a) 求解三角多项式$T_6(x)$。

   (b) 在同一坐标系下,画出图$T_6(x)$和13个数据点。

表5.10　问题5的数据

时　间	温　度	时　间	温　度
1	66	1	58
2	66	2	58
3	65	3	58
4	64	4	58
5	63	5	57
6	63	6	57
7	62	7	57
8	61	8	58
9	60	9	60
10	60	10	64
11	59	11	67
午夜	58	正午	68

表5.11　问题6的数据

日　期	平均温度
1月	-14
1月	-9
2月	2
3月	15
4月	35
5月	52
6月	62
7月	63
8月	58
9月	50
10月	34
11月	12
12月	-5

第6章 数值微分

数值导数的公式对开发求解常微分方程和偏微分方程边值问题的算法很重要(可参见第9章和第10章)。数值微分的例子通常采用已知的函数,这样数值近似值可以与精确解进行比较。为了说明问题,这里采用贝塞耳(Bessel)函数 $J_1(x)$,它的值列表可在标准参考资料中找到。在区间[0,7]内的8个等距点为(0,0.0000),(1,0.4400),(2,0.5767),(3,3391),(4,-0.0660),(5,-0.3276),(6,-0.2767),(7,-0.004)。它所基于的原理是插值多项式的微分。考虑对 $J_1'(2)$ 的求解。插值多项式 $p_2(x)=-0.0710+0.6982x-0.1872x^2$ 经过点(1,0.4400),(2,0.5767),(3,0.3391),而且可用它求出($J_1'(2)\approx p_2'(2)=-0.0505$)。二次多项式 $p_2(x)$ 和它在点(2, $J_1(2)$)的切线如图6.1(a)所示。如果使用5个插值点,可得到更好的近似值。多项式 $p_4(x)=0.4986x+0.011x^2-0.0813x^3+0.0116x^4$ 经过点(0,0.0000),(1,0.4400),(2,0.5767),(3,0.3391),(4,-0.0660),并可用它求出 $J_1'(2)\approx p_4'(2)=-0.0618$。四次多项式 $p_4(x)$ 和它在点(2, $J_1(2)$)的切线如图6.1(b)所示。这个导数的真实值为 $J_1'(2)=-0.0645$,$p_2(x)$ 和 $p_4(x)$ 的误差分别为 -0.0140 和 -0.0026。本章将主要介绍数值微分精确性的相关理论。

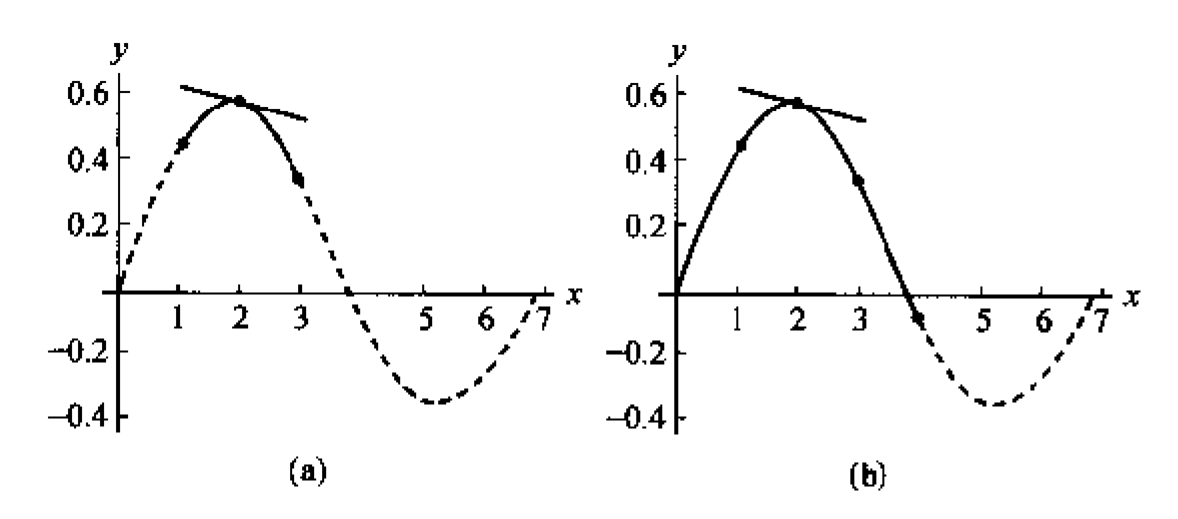


图6.1 (a) $p_2(x)$ 在点(2,0.5767)处的切线,斜率为 $p_2'(2)=-0.0505$
(b) $p_4(x)$ 在点(2,0.5767)处的切线,斜率为 $p'_4(2)=-0.0618$

6.1 导数的近似值

6.1.1 差商的极限

现在研究求解函数 $f(x)$ 的导数近似值的过程,$f(x)$ 的导数可表示为:

$$f'(x)=\lim_{h\to 0}\frac{f(x+h)-f(x)}{h} \tag{1}$$

处理过程非常直接,首先选择一个序列 $\{h_k\}$,使得 $h_k\to 0$,然后计算序列的极限:

$$D_k=\frac{f(x+h_k)-f(x)}{h_k}\qquad k=1,2,\cdots,n,\cdots \tag{2}$$

读者可能注意到上式只计算了序列(2)中有限的项 $D_1, D_2, \cdots, D_N$，而且采用 D_N 作为答案。这带来一些问题，即为什么要计算 $D_1, D_2, \cdots, D_{N-1}$？选择怎样的 h_N，使得 D_N 是导数 $f'(x)$ 较好的近似值？为了回答这些问题，首先看下面的例子，可以发现并没有简单的解决方法。

例如，设有函数 $f(x) = e^x$，并使用步长 $h = 1, 1/2, 1/4$ 分别构造位于 $(0,1)$ 和 $(h, f(h))$ 之间点的割线。当 h 足够小时，割线接近于对应的切线，如图 6.2 所示。尽管图 6.2 给出了(1)中处理过程的可视化表示，但要用 $h = 0.00001$ 才能得到可接受到的答案，采用这个 h 值，图中的切线和割线将无法区分。

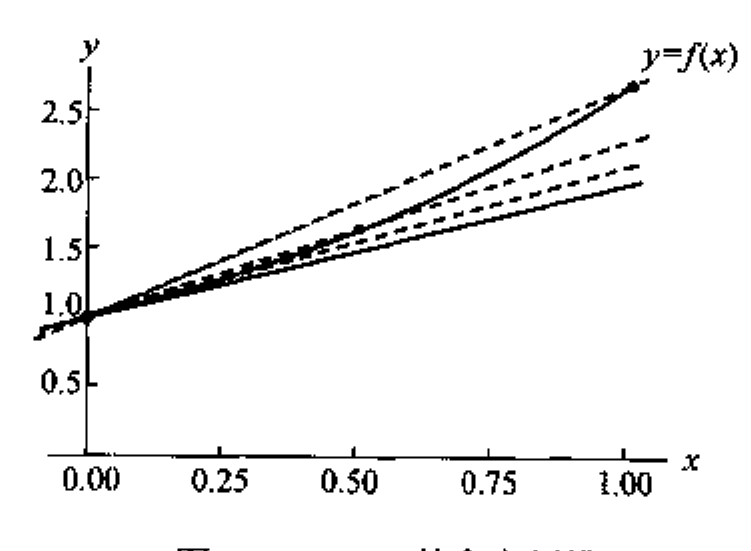


图 6.2　$y = e^x$ 的多个割线

表 6.1　求解 $D_k = (e^{1+h_k} - e)/h_k$ 的差商

h_k	$f_k = f(1+h_k)$	$f_k - e$	$D_k = (f_k - e)/h_k$
$h_1 = 0.1$	3.004166024	0.285884196	2.858841960
$h_2 = 0.01$	2.745601015	0.027319187	2.731918700
$h_3 = 0.001$	2.721001470	0.002719642	2.719642000
$h_4 = 0.0001$	2.718553670	0.000271842	2.718420000
$h_5 = 0.00001$	2.718309011	0.000027183	2.718300000
$h_6 = 10^{-6}$	2.718284547	0.000002719	2.719000000
$h_7 = 10^{-7}$	2.718282100	0.000000272	2.720000000
$h_8 = 10^{-8}$	2.718281856	0.000000028	2.800000000
$h_9 = 10^{-9}$	2.718281831	0.000000003	3.000000000
$h_{10} = 10^{-10}$	2.718281828	0.000000000	0.000000000

例 6.1　设 $f(x) = e^x$ 且 $x = 1$。使用步长 $h_k = 10^{-k}, k = 1, 2, \cdots, 10$ 计算差商 D_k。精度为小数点后 9 位。

计算 D_k 所需的值 $f(1+h_k)$ 和 $(f(1+h_k) - f(1))/h_k$ 如表 6.1 所示。

最大值 $h_1 = 0.1$ 不能得到好的近似值 $D_1 \approx f'(1)$，因为步长 h_1 太大，使得两点分割太远，差商是经过这两点的割线的斜率，不能很好地近似切线。当以小数点后 9 位的精度计算公式(2)时，根据 h_9 可得 $D_9 = 3$，而根据 h_{10} 可得 $D_{10} = 0$。如果 h_k 太小，则 $f(x+h_k)$ 和 $f(x)$ 的值将非常接近。根据差值 $f(x+h_k) - f(x)$ 可看出由于二者太接近，使得精度损失。值 $h_{10} = 10^{-10}$ 太小，使得 $f(x+h_{10})$ 和 $f(x)$ 的值相同，因此计算差商为零。在例 6.1 中，极限的算术值为 $f'(1) \approx 2.718281828$。根据 $h_5 = 10^{-5}$ 可得到最佳近似值 $D_5 = 2.7183$。

例 6.1 显示出不容易求出式(2)中极限的数值近似解。序列在 D_5 时最接近真实值，然后逐渐偏离 e。在程序 6.1 中，当 $|D_{N+1} - D_N| \geq |D_N - D_{N-1}|$ 时，才停止进一步计算序列

$\{D_k\}$中的项。这用来确定在项偏离极限前的最佳近似值。当将这个判定条件用于例6.1时，可得$0.0007 = |D_6 - D_5| > |D_5 - D_4| = 0.00012$，因此答案是$D_5$。下面将研究如何根据较大的h值，得到合理精度的近似值的公式。

6.1.2 中心差分公式

如果函数$f(x)$在点x的左边和右边的值可计算，则最佳二点公式包含x两边的两个对称的横坐标。

定理6.1(精度为$O(h^2)$的中心差分公式) 设$f \in C^3[a,b]$，且$x-h, x, x+h \in [a,b]$，则：

$$f'(x) \approx \frac{f(x+h) - f(x-h)}{2h} \tag{3}$$

而且存在数$c = c(x) \in [a,b]$，满足：

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} + E_{\text{trunc}}(f,h) \tag{4}$$

其中：

$$E_{\text{trunc}}(f,h) = -\frac{h^2 f^{(3)}(c)}{6} = O(h^2)$$

项$E(f,h)$称为截断误差。

证明：设关于x的二阶泰勒展开表达式为$f(x) = P_2(x) + E_2(x)$，则$f(x+h)$和$f(x-h)$的泰勒展开式为：

$$f(x+h) = f(x) + f'(x)h + \frac{f^{(2)}(x)h^2}{2!} + \frac{f^{(3)}(c_1)h^3}{3!} \tag{5}$$

和：

$$f(x-h) = f(x) - f'(x)h + \frac{f^{(2)}(x)h^2}{2!} - \frac{f^{(3)}(C_2)h^3}{3!} \tag{6}$$

式(5)减去式(6)可得：

$$f(x+h) - f(x-h) = 2f'(x)h + \frac{((f^{(3)}(c_1) + f^{(3)}(c_2))h^3}{3!} \tag{7}$$

由于$f^{(3)}(x)$是连续的，所以根据中值定理可找到一个值c，满足：

$$\frac{f^{(3)}(c_1) + f^{(3)}(c_2)}{2} = f^{(3)}(c) \tag{8}$$

将它代入式(7)，并重新调整项可得：

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} - \frac{f^3(c)h^2}{3!} \tag{9}$$

式(9)中右边第一项是中心差分公式(3)，第二项是截断误差。定理得证。

假设三阶导数$f^{(3)}(c)$的值变化不快，则式(4)中的截断误差以与h^2同样的方式趋近于零，表示为$O(h^2)$。当用计算机进行计算时，不宜将h选得太小。为此，如果求解$f'(x)$近似值的公式具有精度为$O(h^4)$的截断误差项，则对于计算机计算很有用。

定理6.2(精度为$O(h^4)$的中心差分公式) 设$f \in C^5[a,b]$，且$x-2h, x-h, x, x+h, x+2h \in [a,b]$，则：

$$f'(x) \approx \frac{-f(x+2h)+8f(x+h)-8f(x-h)+f(x-2h)}{12h} \tag{10}$$

而且存在数 $c=c(x)\in[a,b]$,满足:

$$f'(x)=\frac{-f(x+2h)+8f(x+h)-8f(x-h)+f(x-2h)}{12h}+E_{\text{trunc}}(f,h) \tag{11}$$

其中:

$$E_{\text{trunc}}(f,h)=\frac{h^4 f^{(5)}(c)}{30}=O(h^4)$$

证明:设关于 x 的四阶泰勒展开式为 $f(x)=P_4(x)+E_4(x)$,则 $f(x+h)$ 和 $f(x-h)$ 的泰勒展开式为:

$$f(x+h)-f(x-h)=2f'(x)h+\frac{2f^{(3)}(x)h^3}{3!}+\frac{2f^{(5)}(c_1)h^5}{5!} \tag{12}$$

然后使用步长 2h 代替 h,可得到如下近似值:

$$f(x+2h)-f(x-2h)=4f'(x)h+\frac{16f^{(3)}(x)h^3}{3!}+\frac{64f^{(5)}(c_2)h^5}{5!} \tag{13}$$

式(12)中的项乘以 8 并减去式(13),可消去包含 $f^{(3)}(x)$ 的项,表示为:

$$\begin{aligned}&-f(x+2h)+8f(x+h)-8f(x-h)+f(x-2h)\\&=12f'(x)h+\frac{(16f^{(5)}(c_1)-64f^{(5)}(c_2))h^5}{120}\end{aligned} \tag{14}$$

如果 $f^{(5)}(x)$ 的符号只是正或负,而且它的值变化不快,则可在区间 $[x-2h,x+2h]$ 内找到一个值 c,满足:

$$16f^{(5)}(c_1)-64f^{(5)}(c_2)=-48f^{(5)}(c) \tag{15}$$

将式(15)代入式(14),结果为 $f'(x)$,表示为:

$$f'(x)=\frac{-f(x+2h)+8f(x+h)-8f(x-h)+f(x-2h)}{12h}+\frac{f^{(5)}(c)h^4}{30} \tag{16}$$

式(16)右边的第一项是中心差分公式(10),第二项是截断误差。定理得证。

设 $|f^{(5)}(c)|$ 对于 $c\in[a,b]$ 是有界的,则式(11)中的截断误差以与 h^4 相同的方式趋近于零,表示为 $O(h^4)$。现在比较式(3)和式(10)。设 $f(x)$ 有 5 阶连续导数,而且 $|f^{(3)}(c)|$ 和 $|f^{(5)}(c)|$ 基本相同,则 4 阶公式(10)的截断误差为 $O(h^4)$,2 阶公式(3)的截断误差为 $O(h^2)$,所以式(10)的截断误差比式(3)的截断误差更快地趋近于零。这样在式(10)中可使用更大的步长。

例 6.2 设 $f(x)=\cos(x)$。

(a) 利用式(3)和式(10),步长分别为 $h=0.1,0.01,0.001,0.0001$,计算 $f'(0.8)$ 的近似值。精度为小数点后 9 位。

(b) 与真实值 $f'(0.8)=-\sin(0.8)$ 进行比较。

(a) 设 h = 0.01,根据式(3),可得:

$$f'(0.8)\approx\frac{f(0.81)-f(0.79)}{0.02}\approx\frac{0.689498433-0.703845316}{0.02}\approx-0.717344150$$

设 h = 0.01,根据式(10),可得:

$$f'(0.8)\approx\frac{-f(0.82)+8f(0.81)-8f(0.79)+f(0.78)}{0.12}$$

$$\approx \frac{-0.682221207+8(0.689498433)-8(0.703845316)+0.710913538}{0.12}$$

$$\approx -0.717356108$$

表 6.2 根据式(3)和式(10)得到的数值微分

步长	式(3)的近似值	式(3)的误差	式(10)的近似值	式(10)的误差
0.1	-0.716161095	-0.001194996	-0.717353703	-0.000002389
0.01	-0.717344150	-0.000011941	-0.717356108	0.000000017
0.001	-0.717356000	-0.000000091	-0.717356167	0.000000076
0.0001	-0.717360000	-0.000003909	-0.717360833	0.000004742

(b) 式(3)和式(10)的近似值误差分别为 -0.000011941 和 0.000000017。在本例中,当 h = 0.01 时,式(10)给出的 $f'(0.8)$ 的近似值比式(3)给出的要好。通过对本例的误差分析,可以得出上面的结论。其他的计算如表 6.2 所示。

6.1.3 误差分析和优化步长

关于数值微分的一个重要课题是研究计算机的舍入误差。下面将对此进行更深入的分析。假设使用计算机进行数值计算,而且有:

$$f(x_0-h)=y_{-1}+e_{-1} \text{和} f(x_0+h)=y_1+e_1$$

其中 $f(x_0-h)$ 和 $f(x_0+h)$ 是数值 y_{-1} 和 y_1 的近似值,e_{-1} 和 e_1 分别是相关的舍入误差。下面的结论说明了数值微分中误差分析的复杂性。

推论 6.1(a) 设函数 f 满足定理 6.1 中的假设,并利用计算公式:

$$f'(x_0)\approx\frac{y_1-y_{-1}}{2h} \tag{17}$$

误差分析可通过如下的方程进行解释:

$$f'(x_0)=\frac{y_1-y_{-1}}{2h}+E(f,h) \tag{18}$$

其中:

$$\begin{aligned}E(f,h)&=E_{round}(f,h)+E_{trunc}(f,h)\\&=\frac{e_1-e_{-1}}{2h}-\frac{h^2f^{(3)}(c)}{6}\end{aligned} \tag{19}$$

这里的总误差项 $E(f,h)$ 是舍入误差与截断误差的和。

推论 6.1(b) 设函数 f 满足定理 6.1 的假设,且进行数值计算。如果 $|e_{-1}|\leqslant\varepsilon$, $|e_1|\leqslant\varepsilon$ 且 $M=\max_{a\leqslant x\leqslant b}\{|f^{(3)}(x)|\}$,则:

$$|E(f,h)|\leqslant\frac{\varepsilon}{h}+\frac{Mh^2}{6} \tag{20}$$

式(20)右边最小时的 h 值为:

$$h=\left(\frac{3\varepsilon}{M}\right)^{1/3} \tag{21}$$

当 h 较小时,式(19)中包含 $(e_1-e_{-1})/2h$ 的部分相对较大。在例 6.2 中,当 $h=0.0001$ 时,就会碰到这种情况。舍入误差为:

$$f(0.8001) = 0.696634970 + e_1 \quad , \quad e_1 \approx -0.0000000003$$
$$f(0.7999) = 0.696778442 + e_{-1} \quad , \quad e_{-1} \approx 0.0000000005$$

截断误差为：

$$\frac{-h^2 f^{(3)}(c)}{6} \approx -(0.0001)^2\left(\frac{\sin(0.8)}{6}\right) \approx 0.000000001$$

式(19)中的误差项 $E(f,h)$ 为：

$$E(f,h) \approx \frac{-0.0000000003 - 0.0000000005}{0.0002} - 0.000000001$$
$$= -0.000004001$$

实际上，当 $h = 0.0001$ 时的导数数值近似值可用下式计算：

$$f'(0.8) \approx \frac{f(0.8001) - f(0.7999)}{0.0002} = \frac{0.696634970 - 0.696778442}{0.0002}$$
$$= -0.717360000$$

显然少了4位有效数字。误差是 -0.000003909，接近预计的误差 -0.000004001。

当将式(21)用于例6.2时，可用边界 $|\sin(x)| \leq 1 = M$ 和值 $\varepsilon = 0.5 \times 10^{-9}$ 计算舍入误差。h 的优化值为 $h = (1.5 \times 10^{-9}/1)^{1/3} = 0.001144714$。步长 $h = 0.001$ 时最接近于优化值 0.001144714，而且在包含式(3)的4个选项中，它给出了 $f'(0.8)$ 的最佳近似值(如表6.2和图6.3所示)。

对式(10)的误差分析和上面的类似。设用计算机进行数值计算，并有函数 $f(x_0 + kh) = y_k + e_k$。

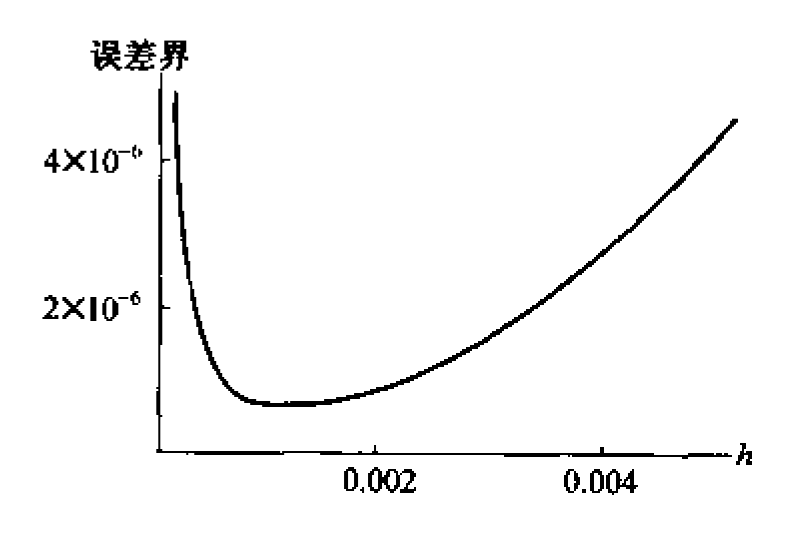


图6.3 将式(21)用于例6.2中的 $f(x) = \cos(x)$ 时，最佳步长为 $h = 0.001144714$

引理6.2(a) 设函数 f 满足定理6.2中的假设，使用计算公式：

$$f'(x_0) \approx \frac{-y_2 + 8y_1 - 8y_{-1} + y_{-2}}{12h} \tag{22}$$

误差分析可用下列方程进行解释：

$$f'(x_0) = \frac{-y_2 + 8y_1 - 8y_{-1} + y_{-2}}{12h} + E(f,h) \tag{23}$$

其中：

$$E(f,h) = E_{\text{round}}(f,h) + E_{\text{trunc}}(f,h)$$

$$= \frac{-e_2 + 8e_1 - 8e_{-1} + e_{-2}}{12h} + \frac{h^4 f^{(5)}(c)}{30} \tag{24}$$

这里的总误差项 $E(f,h)$是舍入误差与截断误差的和。

引理 6.2(b) 设函数 f 满足定理 6.2 的假设,且进行数值计算。如果 $|e_k| \leqslant \varepsilon$ 且 $M = \max_{a \leqslant x \leqslant b}\{|f^{(5)}(x)|\}$,则:

$$|E(f,h)| \leqslant \frac{3\varepsilon}{2h} + \frac{Mh^4}{30} \tag{25}$$

式(25)最小时的 h 值为:

$$h = \left(\frac{45\varepsilon}{4M}\right)^{1/5} \tag{26}$$

当将式(25)用于例 6.2 时,可用边界 $|f^{(5)}(x)| \leqslant |\sin(x)| \leqslant 1 = M$ 和值 $\varepsilon = 0.5\times10^{-9}$ 计算舍入误差。h 的优化值为 $h = (22.5\times10^{-9}/4)^{1/5} = 0.022388475$。步长 $h = 0.01$ 时最接近优化值 0.022388475,而且在包含式(10)的 4 个选项中,给出了 $f'(0.8)$的最佳近似值(如表 6.2 和图 6.4 所示)。

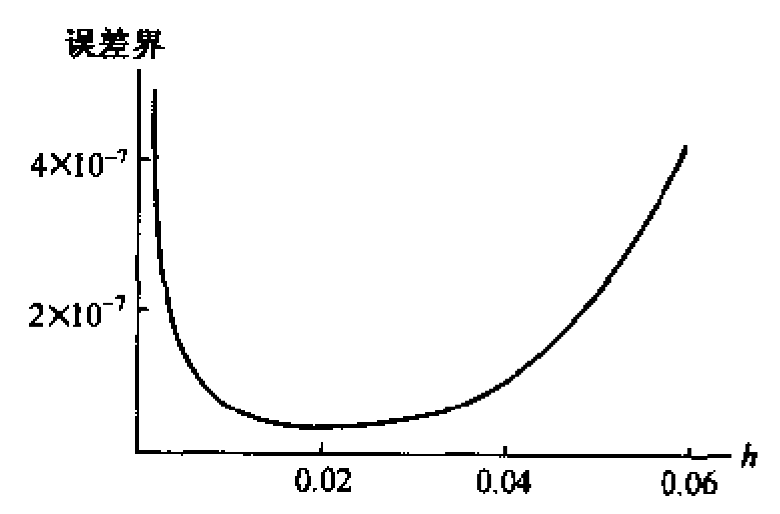


图 6.4 将式(26)用于例 6.2 中的 $f(x) = \cos(x)$ 时,最佳步长为 h = 0.022388475

通过另一种推导也可得到数值微分公式,即对插值多项式进行微分。例如,经过点(0.7, cos(0.7)),(0.8, cos(0.8)),(0.9, cos(0.9))的 2 次多项式 $p_2(x)$的拉格朗日表达式为:

$$p_2(x) = 38.2421094(x-0.8)(x-0.9) - 69.6706709(x-0.7)(x-0.9) + 31.0804984(x-0.7)(x-0.8)$$

将它展开可得:

$$p_2(x) = 1.046875165 - 0.159260044x - 0.348063157x^2$$

通过类似的计算可得到经过点(0.6, cos(0.6)),(0.7, cos(0.7)),(0.8, cos(0.8)),(0.9, cos(0.9)),1.0, cos(1.0))的 4 次多项式 $p_4(x)$:

$$p_4(x) = 0.998452927 + 0.009638391x - 0.523291341x^2 + 0.026521229x^3 + 0.028981100x^4$$

对这些多项式进行微分,可得 $p'_2(0.8) = -0.716161095$ 和 $p'_4(0.8) = -0.717353703$,与表 6.2 中 h = 0.1 下面的值相符。$p_2(x)$和 $p_4(x)$以及它们在点(0.8, cos(0.8))的切线分别如图 6.5(a)和图 6.5(b)所示。

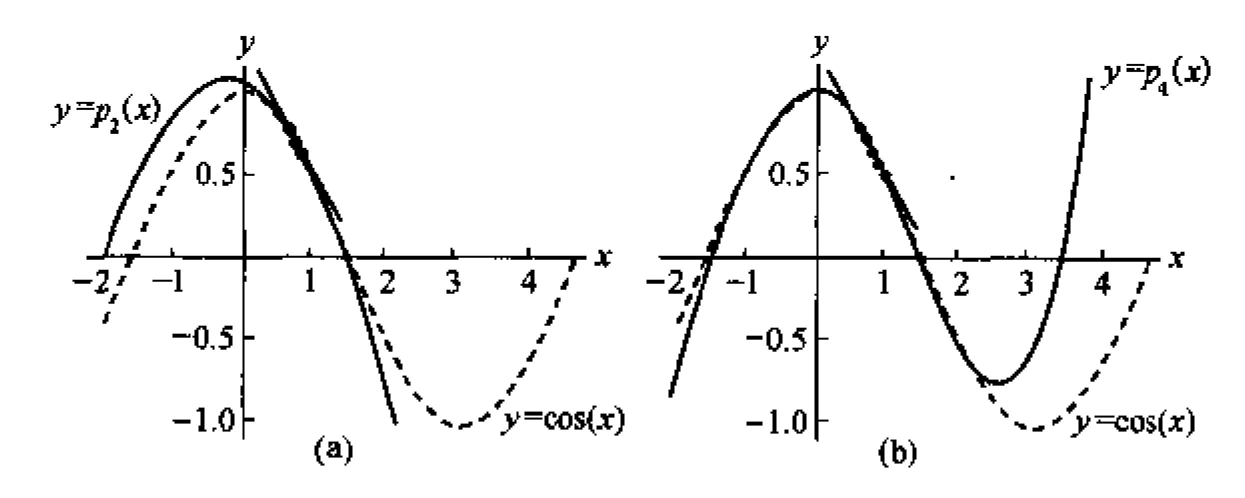


图 6.5 (a) $y=\cos(x)$和用来计算$f'(0.8)\approx p'_2(0.8)=-0.716161095$的插值多项式$p_2(x)$的图形

(b) $y=\cos(x)$和用于计算$f'(0.8)\approx p'_4(0.8)=-0.717353703$的插值多项式$p_4(x)$的图形

6.1.4 Richardson 外推法

这一节将重点研究式(3)与式(10)之间的关系。设$f_k=f(x_k)=f(x_0+kh)$,且用$D_0(h)$和$D_0(2h)$分别表示以h和$2h$为步长,根据式(3)得到的$f'(x_0)$的近似值,表示为:

$$f'(x_0)\approx D_0(h)+Ch^2 \tag{27}$$

和:

$$f'(x_0)\approx D_0(2h)+4Ch^2 \tag{28}$$

如果对式(27)乘以4,并减去式(28),则可消去包含C的项,结果为:

$$3f'(x_0)\approx 4D_0(h)-D_0(2h)=\frac{4(f_1-f_{-1})}{2h}-\frac{f_2-f_{-2}}{4h} \tag{29}$$

对式(29)进一步求解$f'(x_0)$,可得:

$$f'(x_0)\approx\frac{4D_0(h)-D_0(2h)}{3}=\frac{-f_2+8f_1-8f_{-1}+f_{-2}}{12h} \tag{30}$$

式(30)中最后一个表达式是中心差分公式(10)。

例 6.3 设$f(x)=\cos(x)$。$h=0.01$并利用式(27)和式(28),说明式(30)中的线性组合$(4D_0(h)-D_0(2h))/3$如何用来求出式(10)给出的$f'(0.8)$的近似值。精度为小数点后9位。

$h=0.01$并利用式(27)和式(28)可得:

$$D_0(h)\approx\frac{f(0.81)-f(0.79)}{0.02}\approx\frac{0.689498433-0.703845316}{0.02}$$
$$\approx -0.717344150$$

和:

$$D_0(2h)\approx\frac{f(0.82)-f(0.78)}{0.04}\approx\frac{0.682221207-0.710913538}{0.04}$$
$$\approx -0.717308275$$

式(30)中的线性组合为:

$$f'(0.8)\approx\frac{4D_0(h)-D_0(2h)}{3}\approx\frac{4(-0.717344150)-(-0.717308275)}{3}$$
$$\approx -0.717356108$$

这与例6.2中的直接用式(10)得到的$f'(0.8)$的近似值相同。

这种从低阶公式中推导出求解 $f'(x_0)$ 高阶导数的方法称为外推法。相关的证明要求式(3)的误差项可扩展为一个包含 h 的偶次幂的序列。这里已经看到了如何使用步长 h 和 $2h$ 消去包含 h^2 的项。为了说明如何消去 h^4,用 $D_1(h)$ 和 $D_1(2h)$ 分别表示使用步长 h 和 $2h$,根据公式(16)得到的精度为 $f'(x_0)$ 的 $O(h^4)$ 的近似值。则近似值可表示为:

$$f'(x_0)=\frac{-f_2+8f_1-8f_{-1}+f_{-2}}{24h}+\frac{h^4f^{(5)}(c_1)}{30}\approx D_1(h)+Ch^4 \tag{31}$$

和:

$$f'(x_0)=\frac{-f_4+8f_2-8f_{-2}+f_{-4}}{12h}+\frac{16h^4f^{(5)}(c_2)}{30}\approx D_1(2h)+16Ch^4 \tag{32}$$

设 $f^{(5)}(x)$ 只为正或负值,而且变化不快,则可用假设 $f^{(5)}(c_1)\approx f^{(5)}(c_2)$ 来消去式(31)和式(32)中的 h^4,结果为:

$$f'(x_0)\approx\frac{16D_1(h)-D_1(2h)}{15} \tag{33}$$

下面的结论描述了提高计算精度的一般形式。

定理 6.3(Richardson 外推) 设 $f'(x_0)$ 的两个精度为 $O(h^{2k})$ 的近似值分别为 $D_{k-1}(h)$ 和 $D_{k-1}(2h)$,而且它们满足:

$$f'(x_0)=D_{k-1}(h)+c_1h^{2k}+c_2h^{2k+2}+\cdots \tag{34}$$

和:

$$f'(x_0)=D_{k-1}(2h)+4^kc_1h^{2k}+4^{k+1}c_2h^{2k+2}+\cdots \tag{35}$$

这样可得到改进的近似值表达式:

$$f'(x_0)=D_k(h)+O(h^{2k+2})=\frac{4^kD_{k-1}(h)-D_{k-1}(2h)}{4^k-1}+O(h^{2k+2}) \tag{36}$$

下面的程序实现了精度为 $O(h^2)$ 的中心差分公式,即式(3),可得到方程在给定点的导数近似值。在生成的近似值序列 $\{D_k\}$ 中,D_{k+1} 的中心区间是 D_k 的中心区间的十分之一。输出为矩阵 L=[H'D'E'],其中 H 是包含步长的向量,D 是包含导数近似值得向量,E 是包含误差边界的向量。注意:函数 f 作为字符串输入,即'f'。

程序 6.1(使用极限的微分求解) 计算 $f'(x)$ 的近似值,生成序列:

$$f'(x)\approx D_k=\frac{f(x+10^{-k}h)-f(x-10^{-k}h)}{2(10^{-k}h)}\quad k=0,\cdots,n$$

当 $|D_{n+1}-D_n|\geq|D_n-D_{n-1}|$ 或 $|D_n-D_{n-1}|<$ 允许误差时停止计算。后一个不等式用来求最佳近似值 $f'(x)\approx D_n$

```
function  [L,n]=difflim(f,x,toler)
% Input - f is the function input as a string 'f'
%        -x is the differentiation point
%        -toler is the tolerance for the error
% Output-L=[H'D'E']:
%             H is the vector of step sizes
```

```
%             D is the vector of approximate derivatives
%             E is the vector of error bounds
%          -n is the coordinate of the ''best approximation''
max1 = 15;
h = 1;
H(1) = h;
D(1) = (feval(f,x+h) - feval(f,x-h))/(2*h);
E(1) = 0;
R(1) = 0;
for n = 1:2
   h = h/10;
   H(n+1) = h;
   D(n+1) = (feval(f,x+h) - feval(f,x-h))/(2*h);
   E(n+1) = abs(D(n+1) - D(n));
   R(n+1) = 2*E(n+1)*(abs(D(n+1)) + abs(D(n)) + eps);
end
n = 2;
while((E(n) > E(n+1))&(R(n) > toler))&n < max1
   h = h/10;
   H(n+2) = h;
   D(n+2) = (feval(f,x+h) - feval(f,x-h)/(2*h);
   E(n+2) = abs(D(n+2) - D(n+1));
   R(n+2) = 2*E(n+2)*(abs(D(n+2)) + abs(D(n+1)) + eps);
   n = n+1;
end
n = length(D) - 1;
L = [H'D'E'];
```

程序 6.2 实现了定理 6.3(Richardson 外推)。需要注意的是,行 j 中的元素表达式在数学上等价于式(36)。

程序 6.2 (利用外推法的微分求解) 求解 $f'(x)$的数值解,构造包含近似值 $D(j,k)$, $k \leqslant j$ 的表,并将 $f'(x) \approx D(n,n)$作为最终答案。近似值 $D(j,k)$存放在下三角矩阵中。第一列是:

$$D(j,0) = \frac{f(x+2^{-j}h) - f(x-2^{-j}h)}{2^{-j+1}h}$$

行 j 的元素为:

$$D(j,k) = D(j,k-1) + \frac{D(j,k-1) - D(j-1,k-1)}{4^k - 1}, \quad 1 \leqslant k \leqslant j$$

```
function [D,err,relerr,n] = diffext(f,x,delta,toler)
% Input  -f is the function input as a string 'f'
%        - delta is the tolerance for the error
%        - toler is the tolerance for the relative error
% Output - D is the matrix of approximate derivatives
%        - err is the error bound
%        - relerr is the relative error bound
%        - n is the coordinate of the ''best approximation''
```

```
err = 1;
relerr = 1;
h = 1;
j = 1;
D(1,1) = (feval(f,x+h) - feval(f,x-h))/(2*h);
while relerr > toler & err > delta & j < 12
   h = h/2;
   D(j+1,1) = (feval(f,x+h) - feval(f,x-h)/(2*h);
   for k = 1:j
      D(j+1,k+1) = D(j+1,k) + (D(j+1,k) - D(j,k))/((4^k) - 1);
   end
   err = abs(D(j+1,j+1) - D(j,j));
   relerr = 2*err/(abs(D(j+1,j+1)) + abs(D(j,j)) + eps);
   j = j+1;
end
[n,n] = size(D);
```

6.1.5 导数近似值的练习

1. 设 $f(x)=\sin(x)$，x 用弧度表示。
   (a) 步长分别为 $h=0.1, h=0.01, h=0.001$，利用式(3)计算 $f'(0.8)$ 的近似值。精度为小数点后 8 位或 9 位。
   (b) 与值 $f'(0.8)=\cos(0.8)$ 进行比较。
   (c) 计算截断误差(4)的边界。对所有情况使用：
   $$|f^{(3)}(c)| \leqslant \cos(0.7) \approx 0.764842187$$
2. 设 $f(x)=e^x$。
   (a) 步长分别为 $h=0.1, h=0.01, h=0.001$，利用式(3)计算 $f'(2.3)$ 的近似值。精度为小数点后 8 位或 9 位。
   (b) 与值 $f'(2.3)=e^{2.3}$ 进行比较。
   (c) 计算截断误差(4)的边界。对所有情况使用：
   $$|f^{(3)}(c)| \leqslant e^{2.4} \approx 11.02317638$$
3. 设 $f(x)=\sin(x)$，x 用弧度表示。
   (a) 步长分别为 $h=0.1$ 和 $h=0.01$，利用式(10)计算 $f'(0.8)$ 的近似值，并与 $f'(0.8)=\cos(0.8)$ 进行比较。
   (b) 利用式(29)中的外推公式计算(a)中 $f'(0.8)$ 的近似值。
   (c) 计算截断误差(11)的边界。对所有情况使用：
   $$|f^{(5)}(c)| \leqslant \cos(0.6) \approx 0.825335615$$
4. 设 $f(x)=e^x$。
   (a) 步长分别为 $h=0.1$ 和 $h=0.01$，利用式(10)计算 $f'(2.3)$ 的近似值，并与 $f'(2.3)=e^{2.3}$ 进行比较。
   (b) 利用式(29)中的外推公式计算(a)中 $f'(2.3)$ 的近似值。
   (c) 计算截断误差式(11)的边界。对所有情况使用：
   $$|f^{(5)}(c)| \leqslant e^{2.5} \approx 12.18249396$$
5. 比较数值微分公式(3)和(10)。设 $f(x)=x^3$，求解 $f'(2)$ 的近似值。

(a) $h=0.05$，利用式(3)。

(b) $h=0.05$，利用式(10)。

(c) 计算截断误差式(4)和式(11)的边界。

6. (a) 根据泰勒定理证明：

$$f(x+h)=f(x)+hf'(x)+\frac{h^2f^{(2)}(c)}{2},|c-x|<h$$

(b) 根据(a)的结论，证明式(2)中的差商的误差精度为 $O(h)=-hf^{(2)}(c)/2$。

(c) 为何式(3)优于式(2)？

7. 偏微分公式。$f(x,y)$关于 x 的偏导$f_x(x,y)$可通过固定 y 并对 x 求导得到。同理，$f(x,y)$关于 y 的偏导$f_y(x,y)$可通过固定 x 并对 y 求导得到。修改式(3)可得到偏导表达式：

$$f_x(x,y)=\frac{f(x+h,y)-f(x-h,y)}{2h}+O(h^2)$$

$$f_y(x,y)=\frac{f(x,y+h)-f(x,y-h)}{2h}+O(h^2) \tag{i}$$

(a) 设 $f(x,y)=xy/(x+y)$。步长分别为 $h=0.1,0.01$ 和 0.001，用(i)中的公式，计算$f_x(2,3)$和 $f_y(2,3)$的近似值。与通过对 $f(x,y)$求偏导得到的值进行比较。

(b) 设 $z=f(x,y)=\arctan(y/x)$，z 用弧度表示。步长分别为 $h=0.1,0.01,0.001$。用(i)中的公式计算 $f_x(3,4)$和 $f_y(3,4)$的近似值。与通过对 $f(x,y)$进行偏导后得到的结果进行比较。

8. 补充说明根据式(31)和式(32)得到式(33)的细节。

9. (a) 证明式(21)是使式(20)的右边最小化的 h 值。

(b) 证明式(26)是使式(25)的右边最小化的 h 值。

10. 电压值 $E=E(t)$满足关系式 $E(t)=L(dI/dt)+RI(t)$，其中 R 是电阻，L 是电感。设 $L=0.05$，$R=2$，而且 $I(t)$的值如下表所示：

t	$I(t)$
1.0	8.2277
1.1	7.2428
1.2	5.9908
1.3	4.5260
1.4	2.9122

(a) 通过数值微分，求 $I'(1.2)$，并用它计算 $E(1.2)$。

(b) 比较计算结果和 $I(t)=10e^{-t/10}\sin(2t)$。

11. 一个物体的运动距离 $D=D(t)$如下表所示：

t	$D(t)$
8.0	17.453
9.0	21.460
10.0	25.752
11.0	30.301
12.0	35.084

(a) 通过数值微分求速率 $V(10)$。

(b) 比较计算结果和 $D(t)=-70+7t+70e^{-t/10}$。

12. 设 $f(x)$如下表所示。固有的舍入误差的范围为$|e_K|\leqslant 5\times 10^{-6}$。在计算中使用舍入值：

t	$f(x)=\cos(x)$
1.100	0.45360
1.190	0.37166
1.199	0.36329
1.200	0.36236
1.201	0.36143
1.210	0.35302
1.300	0.26750

(a) $h=0.1, h=0.01, h=0.001$，用式(17)求解 $f'(1.2)$的近似值。

(b) 比较计算结果和 $f'(1.2)=-\sin(1.2)\approx -0.93204$。

(c) 针对(a)中的三种情况求解(19)中的总误差界。

13. 设 $f(x)$如下表所示。固有的舍入误差范围为$|e_k|\leqslant 5\times 10^{-6}$。在计算中使用舍入值：

x	$f(x)=\ln(x)$
2.900	1.06471
2.990	1.09527
2.999	1.09828
3.000	1.09861
3.001	1.09895
3.010	1.10194
3.100	1.13140

(a) $h=0.1, h=0.01, h=0.001$，用式(17)求解 $f'(3.0)$的近似值。

(b) 比较计算结果和 $f'(3.0)=\frac{1}{3}\approx 0.33333$。

(c) 针对(a)中的三种情况求解式(19)中的总误差界。

14. 设 $f(x_k)$值的精度为小数点后 3 位，固有舍入误差为 5×10^{-4}。而且假设$|f^{(3)}(c)|\leqslant 1.5$且$|f^{(5)}(c)|\leqslant 1.5$。

(a) 求式(17)的最佳步长 h。

(b) 求式(22)的最佳步长 h。

15. 设 $f(x)$的值如下表所示。固有舍入误差界为$|e_k|\leqslant 5\times 10^{-6}$。在计算中使用舍入值：

x	$f(x)=\cos(x)$
1.000	0.54030
1.100	0.45360
1.198	0.36422
1.199	0.36329
1.200	0.36236
1.201	0.36143
1.202	0.36049
1.300	0.26750
1.400	0.16997

(a) $h=0.1, h=0.001$,用式(22)求$f'(1.2)$的近似值。

(b) 针对(a)中的两种情况求解式(24)中的总误差界。

16. 设$f(x)$的值如下表所示。固有的舍入误差界为$|e_k| \leqslant 5\times 10^{-6}$。在计算中使用舍入值:

x	$f(x)=\ln(x)$
2.800	1.02962
2.900	1.06471
2.998	1.09795
2.999	1.09828
3.000	1.09861
3.001	1.09895
3.002	1.09928
3.100	1.13140
3.200	1.16315

(a) $h=0.1, h=0.001$,用式(22)求$f'(3.0)$的近似值。

(b) 针对(a)中的两种情况求解式(24)中的总误差界。

6.1.6 算法和程序

1. 用程序6.1求解下列函数在x处的导数近似值,精度为小数点后13位。提示:有必要改写程序中max1的值和h的初始值。

   (a) $f(x)=60x^{45}-32x^{33}+233x^5-47x^2-77; x=1/\sqrt{3}$

   (b) $f(x)=\tan\left(\cos\left(\dfrac{\sqrt{5}+\sin x}{1+x^2}\right)\right); x=\dfrac{1+\sqrt{5}}{3}$

   (c) $f(x)=\sin(\cos(1/x)); x=1/\sqrt{2}$

   (d) $f(x)=\sin(x^3-7x^2+6x+8); x=\dfrac{1-\sqrt{5}}{2}$

   (e) $f(x)=x^{x^x}; x=0.0001$

2. 修改程序6.1,实现精度为$O(h^4)$的中心差分公式(10)。用这个程序求解问题1中给出的函数导数的近似值。精度为小数点后13位。

3. 使用程序6.2求解问题1中函数导数的近似值。精度为小数点后13位。提示:有必要改变err,relerr,h的初始值。

6.2 数值差分公式

6.2.1 更多的中心差分公式

在前面的小节中,求解$f'(x_0)$的公式需要可计算函数在x两边的值,所以它们被称为中心差分公式。可用泰勒级数构造求解高阶导数的公式。通常选择精度为$O(h^2)$和$O(h^4)$的公式,如表6.3和表6.4所示。在表中,我们约定对$k=-3,-2,-1,0,1,2,3$,有$f_k=f(x_0+kh)$。

为了说明问题,下面用泰勒展开式推导表6.3中精度为$O(h^2)$的$f''(x)$的公式。其展开式为:

$$f(x+h)=f(x)+hf'(x)+\frac{h^2f''(x)}{2}+\frac{h^3f^{(3)}(x)}{6}+\frac{h^4f^{(4)}(x)}{24}+\cdots \quad (1)$$

表6.3 精度为$O(h^2)$的中心差分公式

$f'(x_0)\approx\frac{f_1-f_{-1}}{2h}$
$f''(x_0)\approx\frac{f_1-2f_0+f_{-1}}{h^2}$
$f^{(3)}(x_0)\approx\frac{f_2-2f_1+2f_{-1}-f_{-2}}{2h^3}$
$f^{(4)}(x_0)\approx\frac{f_2-4f_1+6f_0-4f_{-1}+f_{-2}}{h^4}$

表6.4 精度为$O(h^4)$的中心差分公式

$f'(x_0)\approx\frac{-f_2+8f_1-8f_{-1}+f_{-2}}{12h}$
$f''(x_0)\approx\frac{-f_2+16f_1-30f_0+16f_{-1}-f_{-2}}{12h^2}$
$f^{(3)}(x_0)\approx\frac{-f_3+8f_2-13f_1+13f_{-1}-8f_{-2}+f_{-3}}{8h^3}$
$f^{(4)}(x_0)\approx\frac{-f_3+12f_2-39f_1+56f_0-39f_{-1}+12f_{-2}-f_{-3}}{6h^4}$

和:

$$f(x-h)=f(x)-hf'(x)+\frac{h^2f''(x)}{2}-\frac{h^3f^{(3)}(x)}{6}+\frac{h^4f^{(4)}(x)}{24}-\cdots \quad (2)$$

将式(1)和式(2)相加,将消去包含奇数导数$f'(x),f^{(3)}(x),f^{(5)}(x),\cdots$的项,表示为:

$$f(x+h)+f(x-h)=2f(x)+\frac{2h^2f''(x)}{2}+\frac{2h^4f^{(4)}(x)}{24}+\cdots \quad (3)$$

求解式(3)可得$f''(x)$,表示为:

$$f''(x)=\frac{f(x+h)-2f(x)+f(x-h)}{h^2}-\frac{2h^2f^{(4)}(x)}{4!}-\frac{2h^4f^{(6)}(x)}{6!}-\cdots-\frac{2h^{2k-2}f^{(2k)}(x)}{(2k)!}-\cdots \quad (4)$$

如果将式(4)中的序列在4阶导数处进行截断,则在区间$[x-h,x+h]$内有一个值c,满足:

$$f''(x_0)=\frac{f_1-2f_0+f_{-1}}{h^2}-\frac{h^2f^{(4)}(c)}{12} \quad (5)$$

这样可得到$f''(x)$近似值的公式:

$$f''(x_0)\approx\frac{f_1-2f_0+f_{-1}}{h^2} \quad (6)$$

例6.4 设 $f(x)=\cos(x)$

(a) $h=0.1, 0.01, 0.001$，利用式(6)求解 $f''(0.8)$ 的近似值。精度为小数点后9位。

(b) 比较计算结果和真实值 $f''(0.8)=-\cos(0.8)$。

(a) 当 $h=0.01$ 时，计算过程如下：

$$f''(0.8)\approx\frac{f(0.81)-2f(0.80)+f(0.79)}{0.0001}$$

$$\approx\frac{0.689498433-2(0.696706709)+0.703845316}{0.0001}$$

$$\approx -0.696690000$$

(b) 近似值结果的误差为 -0.000016709。其他的计算如表6.5所示。在误差分析中，将解释在此例中为何 $h=0.01$ 是最佳的。

表6.5 求解例6.4中 $f''(x)$ 的数值近似值

步长	式(6) 得出的近似值	式(6) 产生的误差
$h=0.1$	-0.696126300	-0.000580409
$h=0.01$	-0.696690000	-0.000016709
$h=0.001$	-0.696000000	-0.000706709

6.2.2 误差分析

设 $f_k=y_k+e_k$，其中 e_k 是计算 $f(x_k)$ 产生的误差，包括测量中的噪音和舍入误差，则式(6)可表示为：

$$f''(x_0)=\frac{y_1-2y_0+y_{-1}}{h^2}+E(f,h) \tag{7}$$

式(7)中数值导数的误差项 $E(h,f)$ 包含舍入误差和截断误差：

$$E(f,h)=\frac{e_1-2e_0+e_{-1}}{h^2}-\frac{h^2f^{(4)}(c)}{12} \tag{8}$$

如果设每个误差 e_k 的量级为 ϵ，同时误差是累积的，而且 $|f^{(4)}(x)|\leqslant M$，则可得到如下的误差界：

$$|E(f,h)|\leqslant\frac{4\epsilon}{h^2}+\frac{Mh^2}{12} \tag{9}$$

如果 h 较小，则舍入误差带来的 $4\epsilon/h^2$ 就会较大。当 h 较大，这 $Mh^2/12$ 会较大。可通过求下式的最小值得到最佳步长：

$$g(h)=\frac{4\epsilon}{h^2}+\frac{Mh^2}{12} \tag{10}$$

设 $g'(h)=0$，可得出 $-8\epsilon/h^3+Mh/6=0$，即 $h^4=48\epsilon/M$，这样可得到优化值：

$$h=\left(\frac{48\epsilon}{M}\right)^{1/4} \tag{11}$$

当将式(11)代入例6.4中，使用边界 $|f^{(4)}(x)|\leqslant|\cos(x)|\leqslant 1=M$ 和值 $\epsilon=0.5\times10^{-9}$，可得到优化步长为 $h=(24\times10^{-9}/1)^{1/4}=0.01244666$，而且可看到 $h=0.01$ 时最接近优化值。

由于舍入误差部分与 h 的平方成反比,所以当 h 变小时,这一项会变大。这有时称为步长的两难问题。对此问题的一个部分解决方法是用一个高阶公式,这样可以用较大的 h 值可得到需要精度的近似值。表 6.4 中求精度为 $O(h^4)$的 $f''(x_0)$的公式为:

$$f''(x_0)=\frac{-f_2+16f_1-30f_0+16f_{-1}-f_{-2}}{12h^2}+E(f,h) \tag{12}$$

式(12)中的误差项有如下表达式:

$$E(f,h)=\frac{16\varepsilon}{3h^2}+\frac{h^4f^{(6)}(c)}{90} \tag{13}$$

这里 c 位于区间$[x-2h,x+2h]$。$|E(f,h)|$的界为:

$$|E(f,h)|\leqslant\frac{16\varepsilon}{3h^2}+\frac{h^4M}{90} \tag{14}$$

其中$|f^{(6)}(x)|\leqslant M$。h 的优化值为:

$$h=\left(\frac{240\varepsilon}{M}\right)^{1/6} \tag{15}$$

例 6.5 设 $f(x)=\cos(x)$

(a) $h=1.0,0.1,0.01$,利用式(12)求 $f''(0.8)$的近似值。精度为小数点后 9 位。

(b) 比较计算结果和真实值 $f''(0.8)=-\cos(0.8)$。

(c) 求优化步长。

(a) 当 $h=0.1$ 时,

$$f''(0.8)\approx\frac{-f(1.0)+16f(0.9)-30f(0.8)+16f(0.7)-f(0.6)}{0.12}$$

$$\approx\frac{-0.540302306+9.945759488-20.90120127+12.23747499-0.825335615}{0.12}$$

$$\approx-0.696705958$$

(b) 计算结果的误差为 -0.000000751。其他的计算结果和误差如表 6.6 所示。

(c) 当采用式(15)时,可使用边界$|f^{(6)}(x)|\leqslant|\cos(x)|\leqslant1=M$ 和值$\varepsilon=0.5\times10^{-9}$。根据这些值可得到优化步长 $h=(120\times10^{-9}/1)^{1/6}=0.070231219$。

表 6.6 例 6.5 中求解 $f''(x)$的数值近似值

步　长	式(12) 的近似值	式(12) 的误差
$h=1.0$	-0.689625413	-0.007081296
$h=0.1$	-0.696705958	-0.000000751
$h=0.01$	-0.696690000	-0.000016709

一般来说,如果进行数值微分计算,计算结果只有计算机表示能力的一半精度。除非恰巧找到一个优化步长,否则通常会丢失多位有效数字。所以在进行数值微分计算中,要小心处理。当对精度有限的试验数据进行计算时,困难更大。如果必须根据数据集求数值导数时,应该先用最小二乘法进行曲线拟合,然后对曲线函数进行微分。

6.2.3 拉格朗日多项式微分

如果函数必须在 x_0 的某一边计算，则不能使用中心差分公式。位于 x_0 的右边(左边)的等距横坐标的公式称为前向(后向)差分公式。通过对拉格朗日插值多项式进行差分可得到这些公式。一些常用的前向和后向差分公式如表6.7所示。

表6.7 精度为 $O(h^2)$ 的前向差分公式和后向差分公式

公式	
$f'(x_0)\approx\dfrac{-3f_0+4f_1-f_2}{2h}$	前向微分
$f'(x_0)\approx\dfrac{3f_0-4f_{-1}+f_{-2}}{2h}$	后向微分
$f''(x_0)\approx\dfrac{2f_0-5f_1+4f_2-f_3}{h^2}$	前向微分
$f''(x_0)\approx\dfrac{2f_0-5f_{-1}+4f_{-2}-f_{-3}}{h^2}$	后向微分
$f^{(3)}(x_0)\approx\dfrac{-5f_0+18f_1-24f_2+14f_3-3f_4}{2h^3}$	
$f^{(3)}(x_0)\approx\dfrac{5f_0-18f_{-1}+24f_{-2}-14f_{-3}+3f_{-4}}{2h^3}$	
$f^{(4)}(x_0)\approx\dfrac{3f_0-14f_1+26f_2-24f_3+11f_4-2f_5}{h^4}$	
$f^{(4)}(x_0)\approx\dfrac{3f_0-14f_{-1}+26f_{-2}-24f_{-3}+11f_{-4}-2f_{-5}}{h^4}$	

例6.6 推导公式：

$$f''(x_0)\approx\frac{2f_0-5f_1+4f_2-f_3}{h^2}$$

基于4点 x_0,x_1,x_2,x_3 的 $f(t)$ 的拉格朗日插值多项式为：

$$f(t)\approx f_0\frac{(t-x_1)(t-x_2)(t-x_3)}{(x_0-x_1)(x_0-x_2)(x_0-x_3)}+f_1\frac{(t-x_0)(t-x_2)(t-x_3)}{(x_1-x_0)(x_1-x_2)(x_1-x_3)}$$
$$+f_2\frac{(t-x_0)(t-x_1)(t-x_3)}{(x_2-x_0)(x_2-x_1)(x_2-x_3)}+f_3\frac{(t-x_0)(t-x_1)(t-x_2)}{(x_3-x_0)(x_3-x_1)(x_3-x_2)}$$

对上式求两次导可得：

$$f''(t)\approx f_0\frac{2((t-x_1)+(t-x_2)+(t-x_3))}{(x_0-x_1)(x_0-x_2)(x_0-x_3)}+f_1\frac{2((t-x_0)+(t-x_2)+(t-x_3))}{(x_1-x_0)(x_1-x_2)(x_1-x_3)}$$
$$+f_2\frac{2((t-x_0)+(t-x_1)+(t-x_3))}{(x_2-x_0)(x_2-x_1)(x_2-x_3)}+f_3\frac{2((t-x_0)+(t-x_1)+(t-x_2)}{(x_3-x_0)(x_3-x_1)(x_3-x_2)}$$

用 $t=x_0$ 替换并根据 $x_i-x_j=(i-j)$，可得：

$$f''(x_0)\approx f_0\frac{2((x_0-x_1)+(x_0-x_2)+(x_0-x_3))}{(x_0-x_1)(x_0-x_2)(x_0-x_3)}$$
$$+f_1\frac{2((x_0-x_0)+(x_0-x_2)+(x_0-x_3))}{(x_1-x_0)(x_1-x_2)(x_1-x_3)}$$

$$+f_2\frac{2((x_0-x_0)+(x_0-x_1)+(x_0-x_3))}{(x_2-x_0)(x_2-x_1)(x_2-x_3)}$$

$$+f_3\frac{2((x_0-x_0)+(x_0-x_1)+(x_0-x_2))}{(x_3-x_0)(x_3-x_1)(x_3-x_2)}$$

$$=f_0\frac{2((-h)+(-2h)+(-3h))}{(-h)(-2h)(-3h)}+f_1\frac{2((0)+(-2h)+(-3h))}{(h)(-h)(-2h)}$$

$$+f_2\frac{2((0)+(-h)+(-3h))}{(2h)(h)(-h)}+f_3\frac{2((0)+(-h)+(-2h)}{(3h)(2h)(h)}$$

$$=f_0\frac{-12h}{-6h^3}+f_1\frac{-10h}{2h^3}+f_2\frac{-8h}{-2h^3}+f_3\frac{-6h}{6h^3}=\frac{2f_0-5f_1+4f_2-f_3}{h^2}$$

这样就得到了所需公式。

例6.7 推导公式：

$$f'''(x_0)\approx\frac{-5f_0+18f_1-24f_2+14f_3-3f_4}{2h^3}$$

基于5点 x_0,x_1,x_2,x_3,x_4 的 $f(t)$ 的拉格朗日插值多项式为：

$$f(t)\approx f_0\frac{(t-x_1)(t-x_2)(t-x_3)(t-x_4)}{(x_0-x_1)(x_0-x_2)(x_0-x_3)(x_0-x_4)}$$

$$+f_1\frac{(t-x_0)(t-x_2)(t-x_3)(t-x_4)}{(x_1-x_0)(x_1-x_2)(x_1-x_3)(x_1-x_4)}$$

$$+f_2\frac{(t-x_0)(t-x_1)(t-x_3)(t-x_4)}{(x_2-x_0)(x_2-x_1)(x_2-x_3)(x_2-x_4)}$$

$$+f_3\frac{(t-x_0)(t-x_1)(t-x_2)(t-x_4)}{(x_3-x_0)(x_3-x_1)(x_3-x_2)(x_3-x_4)}$$

$$+f_4\frac{(t-x_0)(t-x_1)(t-x_2)(t-x_3)}{(x_4-x_0)(x_4-x_1)(x_4-x_2)(x_4-x_3)}$$

对上式求3次导，将替换 $x_i-x_j=(i-j)$ 代入分母中可得：

$$f'''(t)\approx f_0\frac{6((t-x_1)+(t-x_2)+(t-x_3)+(t-x_4))}{(-h)(-2h)(-3h)(-4h)}$$

$$+f_1\frac{6((t-x_0)+(t-x_2)+(t-x_3)+(t-x_4))}{(h)(-h)(-2h)(-3h)}$$

$$+f_2\frac{6((t-x_0)+(t-x_1)+(t-x_3)+(t-x_4))}{(2h)(h)(-h)(2h)}$$

$$+f_3\frac{6((t-x_0)+(t-x_1)+(t-x_2)+(t-x_4))}{(3h)(2h)(h)(-h)}$$

$$+f_4\frac{6((t-x_0)+(t-x_1)+(t-x_2)+(t-x_3))}{(4h)(3h)(2h)(h)}$$

在式 $t-x_j=x_0-x_j=-jh$ 中令 $t=x_0$，可得：

$$f'''(x_0)\approx f_0\frac{6((-h)+(-2h)+(-3h)+(-4h))}{24h^4}+f_1\frac{6((0)+(-2h)+(-3h)+(-4h))}{-6h^4}$$

$$+f_2\frac{6((0)+(-h)+(-3h)+(-4h))}{4h^4}+f_3\frac{6((0)+(-h)+(-2h)+(-4h))}{-6h^4}$$

$$+f_4\frac{6((0)+(-h)+(-2h)+(-3h)}{24h^4}$$

$$=f_0\frac{-60h}{24h^4}+f_1\frac{54h}{6h^4}+f_2\frac{-48h}{4h^4}+f_3\frac{42h}{6h^4}+f_4\frac{-36h}{24h^4}$$

$$=\frac{-5f_0+18f_1-24f_2+14f_3-3f_4}{2h^3}$$

这样就得到了所需公式。

6.2.4 牛顿多项式微分

在这一节,将研究用于求 $f'(x_0)$近似值、精度为 $O(h^2)$的3个公式之间的关系,并给出计算数值导数的一般算法。在4.3节中可看到根据点 t_0,t_1,t_2,使用下列2次牛顿多项式 $P(t)$ 近似$f(t)$:

$$P(t)=a_0+a_1(t-t_0)+a_2(t-t_0)(t-t_1) \tag{16}$$

这里 $a_0=f(t_0),a_1=(f(t_1)-f(t_0))/(t_1-t_0)$,且:

$$a_2=\frac{\dfrac{f(t_2)-f(t_1)}{t_2-t_1}-\dfrac{f(t_1)-f(t_0)}{t_1-t_0}}{(t_2-t_0)}$$

$P(t)$的导数为:

$$P'(t)=a_1+a_2((t-t_0)+(t-t_1)) \tag{17}$$

而且当 $t=t_0$ 时,结果为:

$$P'(t_0)=a_1+a_2((t_0-t_1)\approx f'(t_0) \tag{18}$$

用于式(16)到(18)的点集$\{t_k\}$不必是等距的。选择不同的横坐标可导致不同的求解 $f'(x)$近似值的公式。

情况(i):如果 $t_0=x,t_1=x+h,t_2=x+2h$,则:

$$a_1=\frac{f(x+h)-f(x)}{h}$$

$$a_2=\frac{f(x)-2f(x+h)+f(x+2h)}{2h^2}$$

当将这些值代入式(18)中,可得到:

$$P'(x)=\frac{f(x+h)-f(x)}{h}+\frac{-f(x)+2f(x+h)-f(x+2h)}{2h}$$

通过化简可得:

$$P'(x)=\frac{-3f(x)+4f(x+h)-f(x+2h)}{2h}\approx f'(x) \tag{19}$$

这就是 $f'(x)$的二阶前向差分公式。

情况(ii):如果 $t_0=x,t_1=x+h,t_2=x-h$,则:

$$a_1=\frac{f(x+h)-f(x)}{h}$$

$$a_2=\frac{f(x+h)-2f(x)+f(x-h)}{2h^2}$$

当将这些值代入式(18)可得:

$$P'(x)=\frac{f(x+h)-f(x)}{h}+\frac{-f(x+h)+2f(x)-f(x-h)}{2h}$$

通过化简可得:

$$P'(x)=\frac{f(x+h)-f(x-h)}{2h}\approx f'(x) \tag{20}$$

这就是 $f'(x)$ 的二阶中心差分公式。情况(iii):如果 $t_0=x,t_1=x-h,t_2=x-2h$,则:

$$a_1=\frac{f(x)-f(x-h)}{h}$$

$$a_2=\frac{f(x)-2f(x-h)+f(x-2h)}{2h^2}$$

将这些值代入式(18)并进行化简可得:

$$P'(x)=\frac{3f(x)-4f(x-h)+f(x-2h)}{2h}\approx f'(x) \tag{21}$$

这就是 $f'(x)$ 的二阶后向差分公式。

根据点 $t_0,t_1,\cdots,t_N$ 近似 $f(t)$ 的 N 次牛顿多项式表示为:

$$\begin{aligned}P(t)&=a_0+a_1(t-t_0)+a_2(t-t_0)(t-t_1)\\&\quad+a_3(t-t_0)(t-t_1)(t-t_2)+\cdots+a_N(t-t_0)\cdots(t-t_{N-1})\end{aligned} \tag{22}$$

$P(t)$ 的导数为:

$$\begin{aligned}P'(t)&=a_1+a_2((t-t_0)+(t-t_1))\\&\quad+a_3((t-t_0)(t-t_1)+(t-t_0)(t-t_2)+(t-t_1)(t-t_2))\\&\quad+\cdots+a_N\sum_{k=0}^{N-1}\prod_{\substack{j=0\\j\neq k}}^{N-1}(t-t_j)\end{aligned} \tag{23}$$

当在 $t=t_0$ 处计算 $P'(t)$ 时,式中有许多项为零,则 $P'(t_0)$ 可简化为:

$$\begin{aligned}P'(t_0)&=a_1+a_2(t_0-t_1)+a_3(t_0-t_1)(t_0-t_2)+\cdots\\&\quad+a_N(t_0-t_1)(t_0-t_2)(t_0-t_3)\cdots(t_0-t_{N-1})\end{aligned} \tag{24}$$

式(24)右边的第 k 个部分和是根据前 k 个点的 k 次牛顿多项式的导数。如果:

$$|t_0-t_1|\leqslant|t_0-t_2|\leqslant\cdots\leqslant|t_0-t_N|,\text{且 }\{(t_j,0)\}_{j=0}^{N}$$

形成在实数轴上的 N + 1 个等距点,则第 k 个部分和是精度为 $O(h^{k-1})$ 的 $f'(t_0)$ 的近似值。

设 $N=5$。如果有 5 个点 $t_k=x+hk,k=0,1,2,3,4$,则式(24)等价于精度为 $O(h^4)$ 的 $f'(x)$ 的前向差分公式。如果有 5 个点 $t_0=x,t_1=x+h,t_2=x-h,t_3=x+2h,t_4=x-2h$,则式(24)是精度为 $O(h^4)$ 的 $f'(x)$ 的中心差分公式。当 5 个点是 $t_k=x-kh,k=0,1,2,3,4$ 时,式(24)是精度为 $O(h^4)$ 的 $f'(x)$ 的后向差分公式。

下面的程序是程序 4.2 的扩展,可用来实现式(24)。数据点之间不需要等距,而且它只计算一点的导数 $f'(x_0)$。

程序 6.3 (基于 $N+1$ 个点的差分求解) 通过构造下列 N 次牛顿多项式求解 $f'(x)$ 的近似值:

$$P(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1)$$
$$+ a_3(x - x_0)(x - x_1)(x - x_2) + \cdots + a_N(x - x_0)\cdots(x - x_{N-1})$$

将 $f'(x_0) \approx P'(x_0)$ 作为最终结果。在 x_0 处使用这个方法。可通过重新排列点的顺序为 $\{x_k, x_0, \cdots, x_{k-1}, x_{k+1}, \cdots, x_N\}$ 来计算 $f'(x_k) \approx P'(x_k)$

```
function [A,df]=diffnew(X,Y)
% Input  - X is the 1xn abscissa vector
%        - Y is the 1xn ordinate vector
% Output - A is the 1xn vector containing the coefficients of
%          the Nth-degree Newton polynomial
%        - dif is the approximate derivative
A=Y;
N=length(X);
for j=2:N
    for k=N:-1:j
        A(k)=(A(k)-A(k-1))/(X(k)-X(k-j+1));
    end
end
x=X(1);
df=A(2)
prod=1;
n1=length(A)-1;
for k=2:n1
    prod=prod*(x0-X(k));
    df=df+prod*A(k+1);
end
```

6.2.5 数值微分公式的练习

1. 设 $f(x)=\ln(x)$，保证计算精度为小数点后 8 位或 9 位。
   (a) $h=0.05$，利用式(6)计算 $f''(5)$ 的近似值。
   (b) $h=0.01$，利用式(6)计算 $f''(5)$ 的近似值。
   (c) $h=0.1$，利用式(12)计算 $f''(5)$ 的近似值。
   (d) 上述答案中，哪个最精确?
2. 设 $f(x)=\cos(x)$，保证计算精度为小数点后 8 位或 9 位。
   (a) $h=0.05$，利用式(6)计算 $f''(1)$ 的近似值。
   (b) $h=0.01$，利用式(6)计算 $f''(1)$ 的近似值。
   (c) $h=0.1$，利用式(12)计算 $f''(1)$ 的近似值。
   (d) 上述答案中，哪个最精确?

3. 设 $f(x)=\ln(x)$的数据值如下表所示,精度为小数点后 4 位:

x	$f(x)=\ln(x)$
4.90	1.5892
4.95	1.5994
5.00	1.6094
5.05	1.6194
5.10	1.6292

(a) $h=0.05$,利用式(6)计算 $f''(5)$的近似值。
(b) $h=0.01$,利用式(6)计算 $f''(5)$的近似值。
(c) $h=0.05$,利用式(12)计算 $f''(5)$的近似值。
(d) 上述答案中,哪个最精确?

4. 设 $f(x)=\cos(x)$的数据值如下表所示,精度为小数点后 4 位:

x	$f(x)=\cos(x)$
0.90	0.6216
0.95	0.5817
1.00	0.5403
1.05	0.4976
1.10	0.4536

(a) $h=0.05$,利用式(6)计算 $f''(1)$的近似值。
(b) $h=0.01$,利用式(6)计算 $f''(1)$的近似值。
(c) $h=0.05$,利用式(12)计算 $f''(1)$的近似值。
(d) 上述答案中,哪个最精确?

5. $h=0.01$,利用数值微分公式(6)求下列函数的 $f''(1)$的近似值:
(a) $f(x)=x^2$　　(b) $f(x)=x^4$

6. $h=0.1$,利用数值微分公式(12)求下列函数的 $f''(1)$的近似值。
(a) $f(x)=x^4$　　(b) $f(x)=x^6$

7. 对 $f(x+h)$,$f(x-h)$,$f(x+2h)$,$f(x-2h)$进行泰勒展开并推导中心差分公式:

$$f^{(3)}(x)\approx\frac{f(x+2h)-2f(x+h)+2f(x-h)-f(x-2h)}{2h^3}$$

8. 对 $f(x+h)$,$f(x-h)$,$f(x+2h)$,$f(x-2h)$进行泰勒展开并推导中心差分公式:

$$f^{(4)}(x)\approx\frac{f(x+2h)-4f(x+h)+6f(x)-4f(x-h)+f(x-2h)}{h^4}$$

9. 根据下表中的 4 个数据点,求解精度为 $O(h^2)$的 $f'(x_k)$的近似值。

(a)

x	$f(x)$
0.0	0.989992
0.1	0.999135
0.2	0.998295
0.3	0.987480

(b)

x	$f(x)$
0.0	0.141120
0.1	0.041581
0.2	-0.058374
0.3	-0.157746

10. 利用近似值表达式：

$$f'\left(x+\frac{h}{2}\right)\approx\frac{f_1-f_0}{h}\text{和}f'\left(x-\frac{h}{2}\right)\approx\left(\frac{f_0-f_{-1}}{h}\right)$$

推导近似值表达式：

$$f''(x)\approx\frac{f_1-2f_0+f_{-1}}{h^2}$$

11. 根据式(16)到式(18)，基于点 $t_0=x,t_1=x+h,t_2=x+3h$，推导求解 $f'(x)$ 的公式。

12. 根据式(16)到式(18)，基于点 $t_0=x,t_1=x-h,t_2=x+2h$，推导求解 $f'(x)$ 的公式。

13. 特定微分方程的数值解需要精度为 $O(h^2)$ 的 $f''(x)+f'(x)$ 的近似值。
   (a) 通过对精度为 $O(h^2)$ 的 $f'(x)$ 和 $f''(x)$ 求和，求 $f''(x)+f'(x)$ 的中心差分公式。
   (b) 通过对精度为 $O(h^2)$ 的 $f'(x)$ 和 $f''(x)$ 求和，求 $f''(x)+f'(x)$ 的前向差分公式。
   (c) 如果将求解 $f'(x)$ 的精度为 $O(h^4)$ 的公式与求解 $f''(x)$ 的精度为 $O(h^2)$ 的公式相加，情况会怎样？

14. 找出下面论述的问题。通过泰勒公式可得到如下表达式：

$$f(x+h)=f(x)+hf'(x)+\frac{h^2f''(x)}{2}+\frac{h^3f^{(3)}(c)}{6}$$

和：

$$f(x-h)=f(x)-hf'(x)+\frac{h^2f''(x)}{2}-\frac{h^3f^{(3)}(c)}{6}$$

将它们相加可得：

$$f(x-h)+f(x-h)=2f(x)+h^2f''(x).$$

并可得到求解的精确公式：

$$f''(x)=\frac{f(x+h)-2f(x)+f(x-h)}{h^2}$$

6.2.6 算法和程序

1. 修改程序 6.3，使得可以用它计算 $P'(x_M),M=1,2,\cdots,N+1$。

第7章　数值积分

数值积分是工程师和科学家使用的基本工具，用来计算无法解析求解的定积分的近似答案。

在统计热动力学中，计算固体的热效率的德拜(Debye)模型中有如下函数：

$$\Phi(x) = \int_0^x \frac{t^3}{e^t - 1} dt$$

由于不存在 $\Phi(x)$ 的解析表达式，必须用数值积分方法来得到近似值。例如，$\Phi(5)$ 是在 $0 \leqslant t \leqslant 5$ 上曲线 $y = f(t) = t^3/(e^t - 1)$ 之下的面积(见图 7.1)。$\Phi(5)$ 的数值近似为：

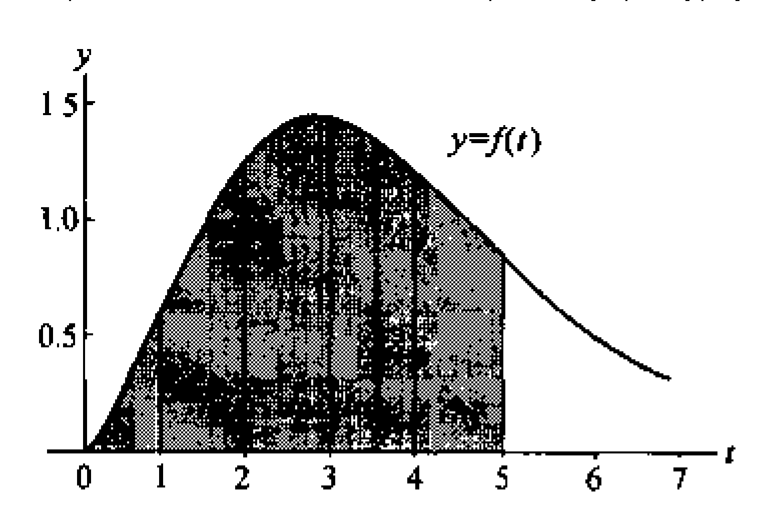


图 7.1　$0 \leqslant t \leqslant 5$ 上曲线 $y = f(t)$ 之下的面积

$\Phi(5)$ 的数值近似为：

$$\Phi(5) = \int_0^5 \frac{t^3}{e^t - 1} dt \approx 4.8998922$$

$\Phi(x)$ 的每个值必须由另一个数值积分求得，表 7.1 列出了在区间[1,10]内的一些近似值。

表 7.1　$\Phi(x)$ 的值

x	$\Phi(x)$
1.0	0.2248052
2.0	1.1763426
3.0	2.5522185
4.0	3.8770542
5.0	4.8998922
6.0	5.5858554
7.0	6.0031690
8.0	6.2396238
9.0	6.3665739
10.0	6.4319219

本章的目的是导出数值积分的基本原理,在第 9 章中,数值积分公式被用来导出微分方程求解的预报 - 校正方法。

7.1 积分简介

数值积分的目的是通过在有限个采样点上计算 $f(x)$ 的值来逼近 $f(x)$ 在区间 $[a,b]$ 内的定积分。

定义 7.1 设 $a = x_0 < x_1 < \cdots < x_M$,形如:

$$Q[f] = \sum_{k=0}^{M} \omega_k f(x_k) = \omega_0 f(x_0) + \omega_1 f(x_1) + \cdots + \omega_M f(x_M) \tag{1}$$

且具有性质:

$$\int_a^b f(x)\,dx = Q[f] + E[f] \tag{2}$$

的公式,称为数值积分或面积公式。项 $E[f]$ 称为积分的截断误差,值 $\{x_k\}_{k=0}^{M}$ 称为面积节点,$\{\omega_k\}_{k=0}^{M}$ 称为权。

根据应用的需要,节点 $\{x_k\}$ 的选择有很多种方法。梯形公式、辛普生公式、以及布尔公式,都选择等距的节点;而高斯 - 勒让德公式中的节点选择为某些勒让德多项式的 0 点。当积分公式用于求微分方程的预报公式时,所有节点都要小于 b。对于任何应用,都需要了解一些关于数值解的精度问题。

定义 7.2 一面积公式的精度为正整数 n,n 使得对所有次数 $i \leqslant n$ 的多项式 $P_i(x)$ 满足 $E(P_i) = 0$,而对某些次数为 $n+1$ 的多项式 $P_{n+1}(x)$ 有 $E[P_{n+1}] \neq 0$。

通过研究当 $f(x)$ 为多项式时的情形可以预测 $E[P_i]$ 的形式,考虑任意 i 次多项式:

$$P_i(x) = a_i x^i + a_{i-1} x^{i-1} + \cdots + a_1 x + a_0$$

若 $i \leqslant n$,则对所有 x,有 $P_i^{(n+1)}(x) \equiv 0$,且 $P_{n+1}^{(n+1)}(x) = (n+1)!a_{n-1}$。因此截断误差的一般形式为:

$$E[f] = Kf^{(n+1)}(c) \tag{3}$$

也就不足为奇了,其中 K 是一合理选择的常数,且 n 为精度。该一般结果的证明可在数值积分的高级教程中找到。

面积公式的推导有时是基于多项式插值的。我们已知,存在惟一的过 $M+1$ 个等距点 $\{(x_k, f(x_k))\}_{k=0}^{M}$ 且次数小于等于 M 的多项式 $P_M(x)$。当用该多项式来近似 $[a,b]$ 内的 $f(x)$ 时,$P_M(x)$ 的积分就近似等于 $f(x)$ 的积分,这一结果的公式称为牛顿 - 柯蒂斯公式(见图 7.2)。当使用采样点 $x_0 = a$ 和 $x_M = b$ 时,称为闭型牛顿 - 柯蒂斯公式。下面的结果给出使用的多项式次数为 $M = 1,2,3,4$ 时的公式。

定理 7.1(闭型牛顿 - 柯蒂斯面积公式) 设 $x_k = x_0 + kh$ 为等距节点,且 $f_k = f(x_k)$。

前 4 个闭型牛顿 - 柯蒂斯面积公式为:

$$\int_{x_0}^{x_1} f(x)\,dx \approx \frac{h}{2}(f_0 + f_1) \quad \text{(梯形公式)} \tag{4}$$

$$\int_{x_0}^{x_2} f(x)\,dx \approx \frac{h}{3}(f_0 + 4f_1 + f_2) \quad (辛普生公式) \tag{5}$$

$$\int_{x_0}^{x_3} f(x)\,dx \approx \frac{3h}{8}(f_0 + 3f_1 + 3f_2 + f_3) \quad (辛普生\frac{3}{8}公式) \tag{6}$$

$$\int_{x_0}^{x_4} f(x)\,dx \approx \frac{2h}{45}(7f_0 + 32f_1 + 12f_2 + 32f_3 + 7f_4) \quad (布尔公式) \tag{7}$$

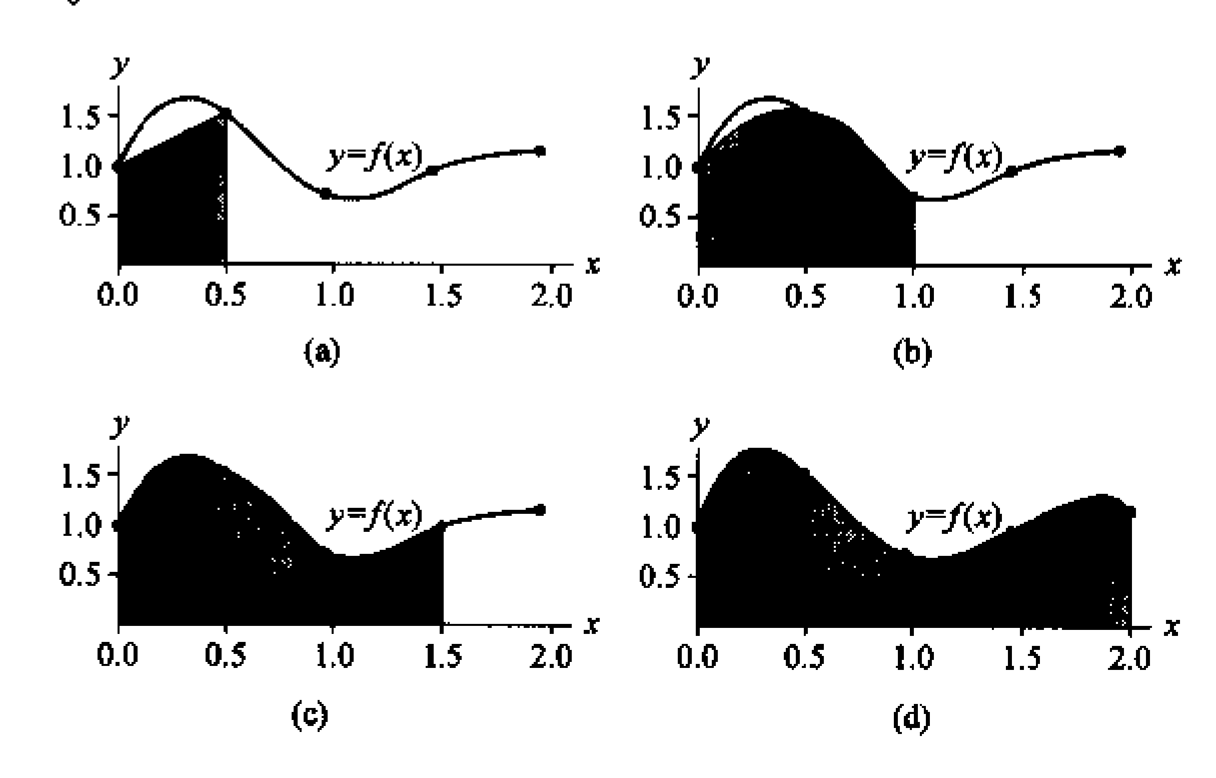


图7.2 (a) $[x_0, x_1] = [0.0, 0.5]$上 $y = P_1(x)$的梯形公式求积分
(b) $[x_0, x_1] = [0.0, 1.0]$上 $y = P_2(x)$的辛普生公式求积分
(c) $[x_0, x_3] = [0.0, 1.5]$上 $y = P_3(x)$的辛普生$\frac{3}{8}$公式求积分
(d) $[x_0, x_4] = [0.0, 2.0]$上 $y = P_4(x)$的布尔公式求积分

推论7.1(牛顿-柯蒂斯公式精度) 设$f(x)$充分可微,则牛顿-柯蒂斯面积公式的 $E[f]$包含一更高阶导数项。梯形公式的精度为 $n = 1$。若 $f \in C^2[a,b]$,则:

$$\int_{x_0}^{x_1} f(x)\,dx = \frac{h}{2}(f_0 + f_1) - \frac{h^3}{12}f^{(2)}(c) \tag{8}$$

辛普生公式的精度为 $n = 3$。若 $f \in C^4[a,b]$,则:

$$\int_{x_0}^{x_2} f(x)\,dx = \frac{h}{3}(f_0 + 4f_1 + f_2) - \frac{h^5}{90}f^{(4)}(c) \tag{9}$$

辛普生$\frac{3}{8}$公式的精度为 $n = 3$。若 $f \in C^4[a,b]$,则:

$$\int_{x_0}^{x_3} f(x)\,dx = \frac{3h}{8}(f_0 + 3f_1 + 3f_2 + f_3) - \frac{3h^5}{80}f^{(4)}(c) \tag{10}$$

布尔公式的精度为 $n = 5$。若 $f \in C^6[a,b]$,则:

$$\int_{x_0}^{x_4} f(x)\,dx = \frac{2h}{45}(7f_0 + 32f_1 + 12f_2 + 32f_3 + 7f_4) - \frac{8h^7}{945}f^{(6)}(c) \tag{11}$$

定理7.1的证明:从基于 $P_M(x)$的 $f(x)$的拉格朗日逼近多项式 $x_0, x_1, \cdots, x_M$ 开始,$f(x)$表示为:

$$f(x) \approx P_M(x) = \sum_{k=0}^{M} f_k L_{M,k}(x) \tag{12}$$

式中 $f_k = f(x_k), k = 0,1,\cdots,M$。得到牛顿－柯蒂斯公式的一般方法：

$$\int_{x_0}^{x_M} f(x)\,dx \approx \int_{x_0}^{x_M} P_M(x)\,dx$$

$$= \int_{x_0}^{x_M}\left(\sum_{k=0}^{M} f_k L_{M,k(x)}\right)dx = \sum_{x=0}^{M}\left(\int_{x_0}^{x_M} f_k L_{M,k(x)}\,dx\right)$$

$$= \sum_{k=0}^{M}\left(\int_{x_0}^{x_M} L_{M,k(x)},\,dx\right)f_k = \sum_{k=0}^{M}\omega_k f_k \tag{13}$$

其中系数 $f_k = f(x_k)$ 的计算细节是枯燥的，我们将给出辛普生公式的一个例证，其中 $M = 2$。这用到逼近多项式：

$$P_2(x) = f_0\,\frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} + f_1\,\frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} + f_2\,\frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} \tag{14}$$

由于 f_0, f_1, f_2 是与积分相关的常数，式(13)变为：

$$\int_{x_0}^{x_2} f(x)\,dx \approx f_0\int_{x_0}^{x_2}\frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)}dx + f_1\int_{x_0}^{x_2}\frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)}dx$$

$$+ f_2\int_{x_0}^{x_2}\frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)}dx \tag{15}$$

右式(15)的积分中引入变量代换用 $dx = hdt$ 代替 $x = x_0 + ht$，新的积分限从 $t = 0$ 到 $t = 2$。根据等距节点 $x_k = x_0 + kh$ 可得 $x_k - x_j = (k-j)h$ 和 $x - x_k = h(t-k)$，这可使(15)式简化为：

$$\int_{x_0}^{x_2} f(x)\,dx \approx f_0\int_0^2\frac{h(t-1)h(t-2)}{(-h)(-2h)}hdt + f_1\int_0^2\frac{h(t-0)h(t-2)}{(h)(-h)}hdt$$

$$+ f_2\int_0^2\frac{h(t-0)h(t-1)}{(2h)(h)}hdt$$

$$= f_0\,\frac{h}{2}\int_0^2(t^2-3t+2)\,dt - f_1 h\int_0^2(t^2-2t)\,dt + f_2\,\frac{h}{2}\int_0^2(t^2-t)\,dt$$

$$= f_0\,\frac{h}{2}\left(\frac{t^3}{3}-\frac{3t^2}{2}+2t\right)\Bigg|_{t=0}^{t=2} - f_1 h\left(\frac{t^3}{3}-t^2\right)\Bigg|_{t=0}^{t=2}$$

$$+ f_2\,\frac{h}{2}\left(\frac{t^3}{3}-\frac{t^2}{2}\right)\Bigg|_{t=0}^{t=2}$$

$$= f_0\,\frac{h}{2}\left(\frac{2}{3}\right) - f_1 h\left(\frac{-4}{3}\right) + f_2\,\frac{h}{2}\left(\frac{2}{3}\right)$$

$$= \frac{h}{3}(f_0 + 4f_1 + f_2) \tag{16}$$

从而证明完毕。在第7.2节中我们将给出推论7.1的一个例证。

例7.1 考虑函数 $f(x) = 1 + e^{-x}\sin(4x)$，等距面积节点为 $x_0 = 0.0, x_1 = 0.5, x_2 = 1.0, x_3 = 1.5$ 和 $x_4 = 2.0$，对应的函数值为 $f_0 = 1.00000, f_1 = 1.55152, f_2 = 0.72159, f_3 = 0.93765$ 和 $f_4 = 1.13390$。利用面积公式(4)～(7)，步长为 $h = 0.5$，计算得：

$$\int_0^{0.5} f(x)\,dx \approx \frac{0.5}{2}(1.00000 + 1.55152) = 0.63788$$

$$\int_0^{1.0} f(x)\,dx \approx \frac{0.5}{3}(1.00000 + 4(1.55152) + 0.72159) = 1.32128$$

$$\int_0^{1.5} f(x)dx \approx \frac{3(0.5)}{8}(1.00000 + 3(1.55152) + 3(0.72159) + 0.93765)$$
$$= 1.64193$$
$$\int_0^{2.0} f(x)dx \approx \frac{2(0.5)}{45}(7(1.00000) + 32(1.55152) + 12(0.72159)$$
$$+ 32(0.93765) + 7(1.13390)) = 2.29444$$

在上面的例子中,式(4)~(7)用来计算不同区间上的定积分的近似值,认识到这一点十分重要。图 7.2(a)~(d)显示了曲线 $y = f(x)$,以及在拉格朗日多项式 $y = P_1(x)$,$y = P_2(x)$,$y = P_3(x)$和 $y = P_4(x)$下的面积。

在例 7.1 中,我们在面积公式中选择了 $h = 0.5$,若区间$[a,b]$的端点固定,则必须对每个公式使用不同的步长。梯形公式、辛普生公式、辛普生$\frac{3}{8}$公式,以及布尔公式的步长分别为 $h = b - a$,$h = (b - a)/2$,$h = (b - a)/3$ 和 $h = (b - a)/4$。下面的例子显示了这一点。

例 7.2 考虑函数 $f(x) = 1 + e^{-x}\sin(4x)$在固定区间$[a,b] = [0,1]$内的积分。使用式(4)~(7)来计算其值。

解:

对梯形公式,$h = 1$,且:
$$\int_0^1 f(x)dx \approx \frac{1}{2}(f(0) + f(1))$$
$$= \frac{1}{2}(1.00000 + 0.72159) = 0.86079$$

对辛普生公式,$h = 1/2$,且:
$$\int_0^1 f(x)dx \approx \frac{1/2}{3}\left(f(0) + 4f\left(\frac{1}{2}\right) + f(1)\right)$$
$$= \frac{1}{6}(1.00000 + 4(1.55152) + 0.72159) = 1.32128$$

对辛普生$\frac{3}{8}$公式,$h = 1/3$,而:
$$\int_0^1 f(x)dx \approx \frac{3(1/3)}{8}\left(f(0) + 3f\left(\frac{1}{3}\right) + 3f\left(\frac{2}{3}\right) + f(1)\right)$$
$$= \frac{1}{8}(1.00000 + 3(1.69642) + 3(1.23447) + 0.72159) = 1.31440$$

对布尔公式,$h = 1/4$,结果为:
$$\int_0^1 f(x)dx \approx \frac{2(1/4)}{45}\left(7f(0) + 32f\left(\frac{1}{4}\right) + 12f\left(\frac{1}{2}\right) + 32f\left(\frac{3}{4}\right) + 7f(1)\right)$$
$$= \frac{1}{90}(7(1.00000) + 32(1.65534) + 12(1.55152))$$
$$+ 32(1.06666) + 7(0.72159)) = 1.30859$$

该定积分的真解为:
$$\int_0^1 f(x)dx = \frac{21e - 4\cos(4) - \sin(4)}{17e} = 1.3082506046426\cdots$$

布尔公式的近似结果最好。图 7.3(a)～(d)分别给出了拉格朗日多项式 $P_1(x)$，$P_2(x)$，$P_3(x)$和 $P_4(x)$下的面积。

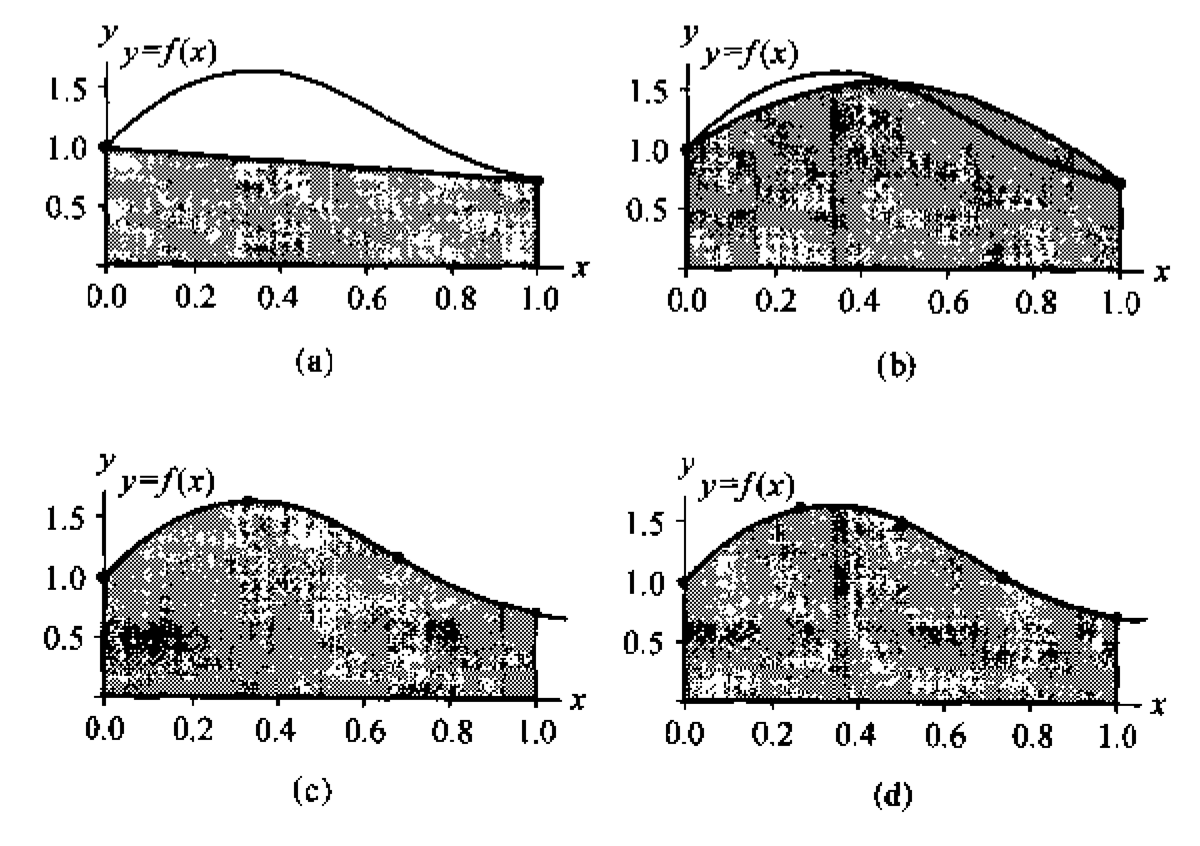


图 7.3　(a)区间[0,1]内的梯形公式结果为 0.86079
(b)区间[0,1]内的辛普生公式结果为 1.32128
(c)区间[0,1]内的辛普生$\frac{3}{8}$公式结果为 1.31440
(d)区间[0,1]内的布尔公式结果为 1.30859

要对面积公式进行公平的比较，必须在每种方法中使用相同数目的函数求值。最后一个例子比较了给定区间[a,b]内的积分，每种方法中使用 5 个函数求值 $f_k=f(x_k)$，$k=0,1,\ldots,4$。当在 4 个子区间$[x_0,x_1]$，$[x_1,x_2]$，$[x_2,x_3]$和$[x_3,x_4]$上使用梯形公式时，称之为组合梯形公式：

$$\begin{aligned}\int_{x_0}^{x_4} f(x)\,dx &= \int_{x_0}^{x_1} f(x)\,dx + \int_{x_1}^{x_2} f(x)\,dx + \int_{x_2}^{x_3} f(x)\,dx + \int_{x_3}^{x_4} f(x)\,dx \\ &\approx \frac{h}{2}(f_0+f_1)+\frac{h}{2}(f_1+f_2)+\frac{h}{2}(f_2+f_3)+\frac{h}{2}(f_3+f_4) \\ &= \frac{h}{2}(f_0+2f_1+2f_2+2f_3+f_4)\end{aligned} \tag{17}$$

对辛普生公式也可使用相同的方法，当在两个子区间$[x_0,x_2]$和$[x_2,x_4]$上使用辛普生公式时，称之为组合辛普生公式：

$$\begin{aligned}\int_{x_0}^{x_4} f(x)\,dx &= \int_{x_0}^{x_2} f(x)\,dx + \int_{x_2}^{x_4} f(x)\,dx \\ &\approx \frac{h}{3}(f_0+4f_1+f_2)+\frac{h}{3}(f_2+4f_3+f_4) \\ &= \frac{h}{3}(f_0+4f_1+2f_2+4f_3+f_4)\end{aligned} \tag{18}$$

下面的例子比较了由(17)、(18)和(7)式得到的结果。

例 7.3　考虑函数 $f(x)=1+e^{-x}\sin(4x)$ 在区间 $[a,b]=[0,1]$ 内的积分，使用 5 次函数求值，并比较组合梯形公式，组合辛普生公式，以及布尔公式的结果。

解：

相同的步长为 $h=1/4$，组合梯形公式(17)得到：

$$\int_0^1 f(x)\,dx \approx \frac{1/4}{2}\left(f(0)+2f(\frac{1}{4})+2f(\frac{1}{2})+2f(\frac{3}{4})+f(1)\right)$$

$$=\frac{1}{8}(1.00000+2(1.65534)+2(1.55152)+2(1.06666)+0.72159)$$

$$=1.28358$$

由组合辛普生公式(18)，得：

$$\int_0^1 f(x)\,dx \approx \frac{1/4}{3}\left(f(0)+4f\left(\frac{1}{4}\right)+2f\left(\frac{1}{2}\right)+4f\left(\frac{3}{4}\right)+f(1)\right)$$

$$=\frac{1}{12}(1.00000+4(1.65534)+2(1.55152)+4(1.06666)+0.72159)$$

$$=1.30938$$

布尔公式的结果在例 7.2 中已经求得，为：

$$\int_0^1 f(x)\,dx \approx \frac{2(1/4)}{45}\left(7f(0)+32f\left(\frac{1}{4}\right)+12f\left(\frac{1}{2}\right)+32f\left(\frac{3}{4}\right)+7f(1)\right)$$

$$=1.30859$$

积分的真解为：

$$\int_0^1 f(x)\,dx = \frac{21e-4\cos(4)-\sin(4)}{17e} = 1.3082506046426\cdots$$

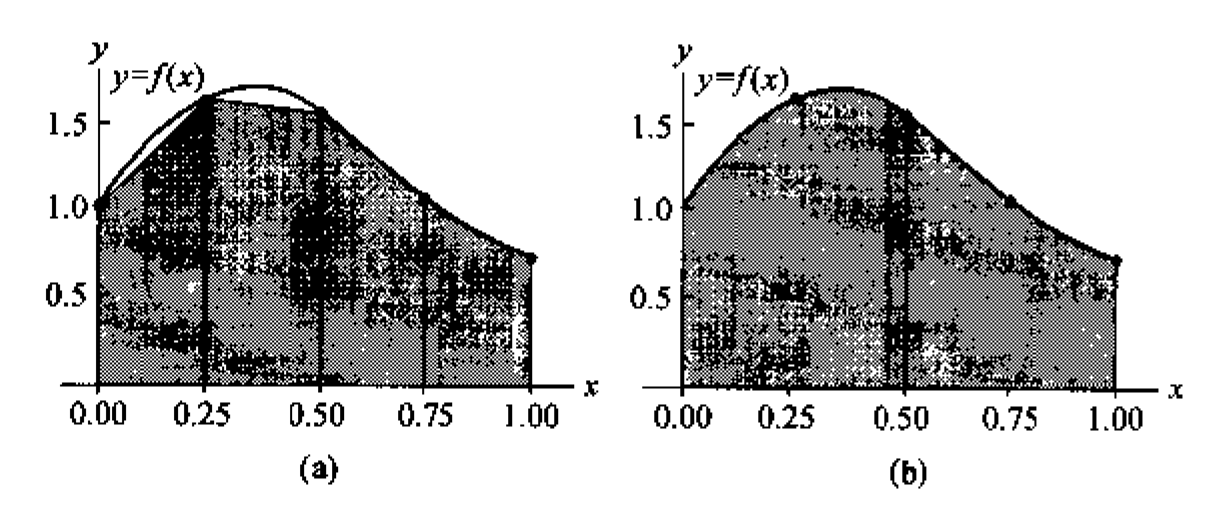


图 7.4　(a)组合梯形的结果为 1.28358；(b)组合辛普生公式的结果为 1.30938

由辛普生公式所得的结果 1.30938 要优于梯形公式得到的 1.28358；而布尔公式的结果 1.30859 仍是最接近真值的。图 7.4(a)和(b)分别给出了梯形和抛物线下的面积。

例 7.4　求辛普生 $\frac{3}{8}$ 公式的精度。

解：

在区间 $[0,3]$ 内对 5 个测试函数 $f(x)=1,x,x^2,x^3$ 和 x^4。应用辛普生 $\frac{3}{8}$ 公式足以说明问题。对前 4 个函数，辛普生 $\frac{3}{8}$ 公式是精确的：

$$\int_0^3 1dx = 3 = \frac{3}{8}(1 + 3(1) + 3(1) + 1)$$

$$\int_0^3 xdx = \frac{9}{2} = \frac{3}{8}(0 + 3(1) + 3(2) + 3)$$

$$\int_0^3 x^2 dx = 9 = \frac{3}{8}(0 + 3(1) + 3(4) + 9)$$

$$\int_0^3 x^3 dx = \frac{81}{4} = \frac{3}{8}(0 + 3(1) + 3(8) + 27)$$

函数 $f(x)=x^4$ 是最低次的 x 幂函数，使得该公式不精确：

$$\int_0^3 x^4 dx = \frac{243}{5} \approx \frac{99}{2} = \frac{3}{8}(0 + 3(1) + 3(16) + 81)$$

故辛普生 $\frac{3}{8}$ 公式的精度为 $n=3$。

7.1.1 习题

1. 使用面积公式(4) ~ (7)计算函数 $f(x)$ 在固定区间 $[a,b]=[0,1]$ 内的积分。梯形公式、辛普生公式、辛普生 $\frac{3}{8}$ 公式、布尔公式的步长分别为 $h=1, h=\frac{1}{2}, h=\frac{1}{3}$ 和 $h=\frac{1}{4}$。
   函数 $f(x)$ 为：
   (a) $f(x)=\sin(\pi x)$
   (b) $f(x)=1+e^{-x}\cos(4x)$
   (c) $f(x)=\sin(\sqrt{x})$
   注：定积分的真解为：(a) $2/\pi=0.636619772367\cdots$，(b) $(18e-\cos(4)+4\sin(4))/(17e)=1.007459631397\cdots$，(c) $2(\sin(1)-\cos(1)=0.602337357879\cdots$。函数的曲线分别在图 7.5(a) ~ (c)中给出。

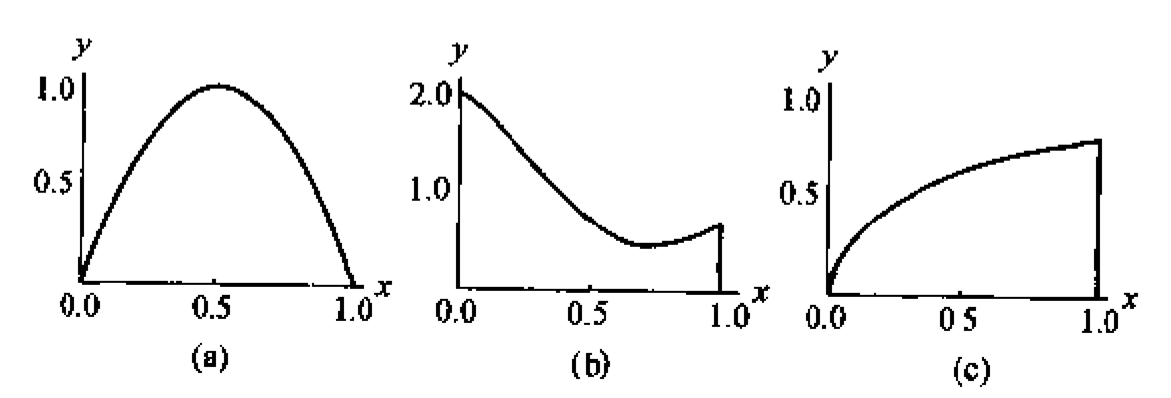


图 7.5　(a) $y=\sin(\pi x)$；(b) $y=1+e^{-x}\cos(4x)$；(c) $y=\sin\sqrt{x}$

2. 计算函数 $f(x)$ 在固定区间 $[a,b]=[0,1]$ 内的积分。应用组合梯形公式(17)、组合辛普生公式(18)以及布尔公式(7)，使用 5 个等距节点上的函数值，步长为 $h=\frac{1}{4}$。
   函数 $f(x)$ 为：
   (a) $f(x)=\sin(\pi x)$
   (b) $f(x)=1+e^{-x}\cos(4x)$
   (c) $f(x)=\sin(\sqrt{x})$
3. 考虑一般的区间 $[a,b]$，证明辛普生公式对函数 $f(x)=x^2$ 和 $f(x)=x^3$ 有精确解，

即：

(a) $\int_a^b x^2\,dx = \frac{b^3}{3} - \frac{a^3}{3}$ (b) $\int_a^b x^3\,dx = \frac{b^4}{4} - \frac{a^4}{4}$

4. 在区间上$[x_0, x_1]$求拉格朗日插值多项式：

$$P_1(x) = f_0\frac{x - x_1}{x_0 - x_1} + f_1\frac{x - x_0}{x_1 - x_0}$$

的积分，并建立梯形公式。

5. 求梯形公式的精度。只需对[0,1]内的 3 个函数 $f(x) = 1, x$ 和 x^2 应用梯形公式即可。

6. 求辛普生公式的精度，只需对[0,2]内的 5 个测试函数 $f(x) = 1, x, x^2, x^3$ 和 x^4 应用辛普生公式即可。将结果与辛普生$\frac{3}{8}$公式的精度相比较。

7. 求布尔公式的精度，只需对[0,4]内的 7 个函数 $f(x) = 1, x, x^2, x^3, x^4, x^5$ 和 x^6 应用布尔公式即可。

8. 习题 5，习题 6，习题 7 和例 7.4 中的区间选择是为了简化面积节点的计算，但在任何函数 f 可积的闭区间$[a,b]$内，式(4)～(7)的精度与习题 5～7 及例 7.4 所求的精度相同。区间$[a,b]$内的面积公式可由区间$[c,d]$内的面积公式通过如下的变量代换得到：

$$x = g(t) = \frac{b - a}{d - c}t + \frac{ad - bc}{d - c}$$

其中 $dx = \frac{b - a}{d - c}dt$。

(a) 证明：$x = g(t)$是过点(c, a)和(d, b)的一条直线。

(b) 证明：梯形公式在区间[a,b]和区间[0,1]内有相同的精度。

(c) 证明：辛普生公式在区间$[a,b]$和区间[0,2]内有相同的精度。

(d) 证明：布尔公式在区间$[a,b]$和区间[0,4]内有相同的精度。

9. 用拉格朗日多项式插值推导辛普生$\frac{3}{8}$公式。提示：变量代换后，可得到与(16)式相近的积分：

$$\begin{aligned}\int_{x_0}^{x_3} f(x)\,dx \approx & -f_0\frac{h}{6}\int_0^3 (t-1)(t-2)(t-3)\,dt + f_1\frac{h}{2}\int_0^3 (t-0)(t-2)(t-3)\,dt \\ & - f_2\frac{h}{2}\int_0^3 (t-0)(t-1)(t-3)\,dt + f_3\frac{h}{6}\int_0^3 (t-0)(t-1)(t-2)\,dt \\ = & f_0\frac{h}{6}\left(\frac{-t^4}{4} + 2t^3 - \frac{11t^2}{2} + 6t\right)\Bigg|_{t=0}^{t=3} + f_1\frac{h}{2}\left(\frac{t^4}{4} - \frac{5t^3}{3} + 3t^2\right)\Bigg|_{t=0}^{t=3} \\ & + f_2\frac{h}{2}\left(\frac{-t^4}{4} + \frac{4t^3}{3} - \frac{3t^2}{2}\right)\Bigg|_{t=0}^{t=3} + f_3\frac{h}{6}\left(\frac{t^4}{4} - t^3 + t^2\right)\Bigg|_{t=0}^{t=3}\end{aligned}$$

10. 基于 5 次拉格朗日逼近多项式，用 6 个等距节点 $x_k = x_0 + kh, k = 0, 1, \cdots, 5$，推导闭型牛顿－柯蒂斯公式。

11. 在定理 7.1 的证明中，辛普生公式由基于 3 个等距节点 x_0, x_1 和 x_2 的 2 次拉格朗日多项式的积分推导。试用基于 3 个等距节点 x_0, x_1 和 x_2 的 2 次牛顿多项式的积分推导出辛普生公式。

7.2 组合梯形公式和辛普生公式

求$[a,b]$内曲线 $y=f(x)$下的面积的直观方法，是用区间$\{[x_k,x_{k+1}]\}$上的一系列梯形的面积来近似。

定理 7.2(组合梯形公式) 设区间$[a,b]$被等距节点 $x_k=a+kh,k=0,1,\cdots,M$,分为宽度为 $h=(b-a)/M$ 的M 个子区间$[x_k,x_{k+1}]$。M 个子区间的组合梯形公式可以表示为三种等价方式之中的任何一种：

$$T(f,h)=\frac{h}{2}\sum_{k=1}^{M}(f(x_{k-1})+f(x_k)) \tag{1a}$$

或：

$$T(f,h)=\frac{h}{2}(f_0+2f_1+2f_2+2f_3+\cdots+2f_{M-2}+2f_{M-1}+f_M) \tag{1b}$$

或：

$$T(f,h)=\frac{h}{2}(f(a)+f(b))+h\sum_{k=1}^{M-1}f(x_k) \tag{1c}$$

这是区间$[a,b]$内 $f(x)$积分的一种逼近，写为：

$$\int_a^b f(x)\,dx\approx T(f,h) \tag{2}$$

证明：在每个子区间$[x_{k-1},x_k]$上应用梯形公式(见图 7.6)，利用子区间上积分的可加性：

$$\int_a^b f(x)\,dx=\sum_{k=1}^{M}\int_{x_{k-1}}^{x_k}f(x)\,dx\approx\sum_{k=1}^{M}\frac{h}{2}(f(x_{k-1})+f(x_k)) \tag{3}$$

由于 $h/2$ 为常数，由加法分配律可得式(1a)。式(1b)为式(1a)的展开式。式(1c)表示如何将式(1b)中所有被 2 乘的项写在一起的形式。

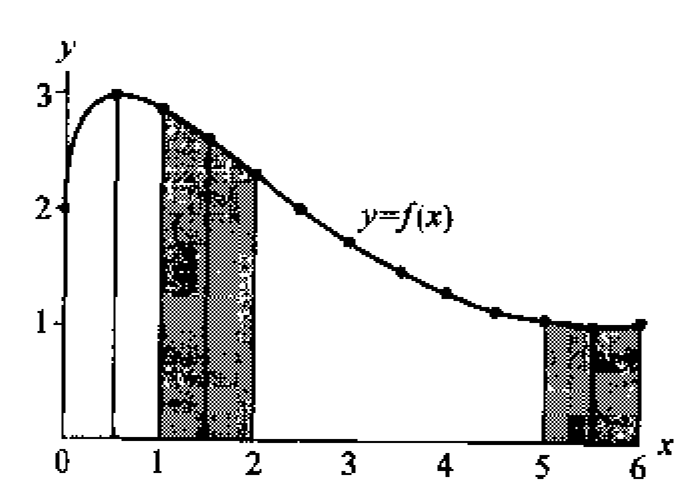


图 7.6 用组合梯形公式逼近曲线 $y=2+\sin(2\sqrt{x})$下的面积

用分段线性多项式逼近函数 $f(x)=2+\sin(2\sqrt{x})$，得到的结果中有些点处为闭形式的逼近，有些点处为开形式的逼近。为了保证精度，必须在许多子区间上应用组合梯形公式。在下面的例子中我们在区间[1,6]内进行数值积分，区间[0,1]内的积分留作练习。

例 7.5 考虑函数 $f(x)=2+\sin(2\sqrt{x})$，利用组合梯形公式和 11 个采样点来计算区间[1,6]内 $f(x)$ 的积分的近似值。

解：

我们用 $M=10$ 和 $h=(6-1)/10=1/2$ 生成 11 个采样点，利用式(1c)，计算得：

$$
\begin{aligned}
T\left(f,\frac{1}{2}\right) &= \frac{1/2}{2}(f(1)+f(6)) \\
&+\frac{1}{2}\left(f\left(\frac{3}{2}\right)+f(2)+f\left(\frac{5}{2}\right)+f(3)+f\left(\frac{7}{2}\right)+f(4)+f\left(\frac{9}{2}\right)+f(5)+f\left(\frac{11}{2}\right)\right) \\
&=\frac{1}{4}(2.90929743+1.01735756) \\
&+\frac{1}{2}(2.63815764+2.30807174+1.97931647+1.68305284+1.43530410 \\
&+1.24319750+1.10831775+1.02872220+1.00024140) \\
&=\frac{1}{4}(3.92665499)+\frac{1}{2}(14.42438165) \\
&=0.98166375+7.21219083=8.19385457
\end{aligned}
$$

定理 7.3(组合辛普生公式) 设区间 $[a,b]$ 由 $x_k=a+kh,k=0,1,\cdots,2M$ 分为 $2M$ 个宽度为 $h=(b-a)/2M$ 的等距子区间 $[x_k,x_{k+1}]$。$2M$ 个子区间的组合辛普生公式可表示为三种等价方式之一：

$$S(f,h)=\frac{h}{3}\sum_{k=1}^{M}(f(x_{2k-2})+4f(x_{2k-1})+f(x_{2k})) \tag{4a}$$

或：

$$
\begin{aligned}
S(f,h)=\frac{h}{3}(f_0+4f_1+2f_2+4f_3 \\
+\cdots+2f_{2M-2}+4f_{2M-1}+f_{2M})
\end{aligned}
\tag{4b}
$$

或：

$$S(f,h)=\frac{h}{3}(f(a)+f(b))+\frac{2h}{3}\sum_{k=1}^{M-1}f(x_{2k})+\frac{4h}{3}\sum_{k=1}^{M}f(x_{2k-1}) \tag{4c}$$

这是对 $f(x)$ 在 $[a,b]$ 内积分的一种逼近，写为：

$$\int_a^b f(x)\,dx \approx S(f,h) \tag{5}$$

证明：在每个子区间 $[x_{2k-2},x_{2k}]$ 上应用辛普生公式(见图 7.7)，利用子区间上积分的可加性：

$$
\begin{aligned}
\int_a^b f(x)\,dx &= \sum_{k=1}^{M}\int_{x_{2k-2}}^{x_{2k}} f(x)\,dx \\
&\approx \sum_{k=1}^{M}\frac{h}{3}(f(x_{2k-2})+4f(x_{2k-1})+f(x_{2k}))
\end{aligned}
\tag{6}
$$

由于 $h/3$ 为常数，可由加法分配律得到式(4a)，式(4b)是式(4a)的展开形式。式(4c)是将式(4b)中的被 2 乘和被 4 乘的项写在一起的形式。

用分段 2 次多项式逼近函数 $f(x)=2+\sin(2\sqrt{x})$，得到逼近为闭的和非闭的形式，为了保证精度，必须在许多子区间上应用组合辛普生公式。在下面的例子中我们在区间[1,6]内进行数值积分，区间[0,1]内的积分留作练习。

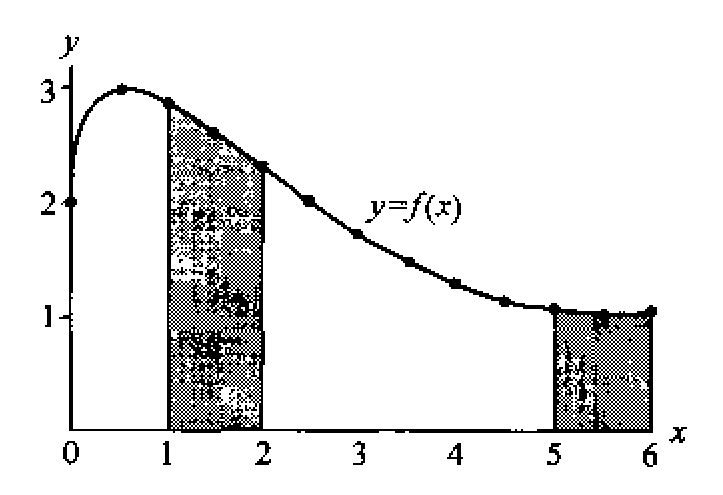


图 7.7　用组合辛普生公式逼近曲线 $y=2+\sin(2\sqrt{x})$ 下的面积

例 7.6　考虑函数 $f(x)=2+\sin(2\sqrt{x})$，利用组合公式和 11 个采样点来计算区间 $[1,6]$ 内的 $f(x)$ 的积分的近似值。

解：

我们用 $M=5$ 和 $h=(6-1)/10=1/2$ 生成 11 个采样点，利用式(4c)，计算得：

$$
\begin{aligned}
S(f,\tfrac{1}{2}) &= \frac{1}{6}(f(1)+f(6))+\frac{1}{3}(f(2)+f(3)+f(4)+f(5)) \\
&\quad +\frac{2}{3}\left(f\left(\frac{3}{2}\right)+f\left(\frac{5}{2}\right)+f\left(\frac{7}{2}\right)+f\left(\frac{9}{2}\right)+f\left(\frac{11}{2}\right)\right) \\
&= \frac{1}{6}(2.90929743+1.01735756) \\
&\quad +\frac{1}{3}(2.30807174+1.68305284+1.24319750+1.02872220) \\
&\quad +\frac{2}{3}(2.63815764+1.97931647+1.43530410+1.10831775+1.00024140) \\
&= \frac{1}{6}(3.92665499)+\frac{1}{3}(6.26304429)+\frac{2}{3}(8.16133735) \\
&= 0.65444250+2.08768143+5.44089157=8.18301550
\end{aligned}
$$

7.2.1　误差分析

下面两个结果的重要性在于，知道了组合梯形公式和组合辛普生公式的误差项 $E_T(f,h)$ 和 $E_S(f,h)$ 分别为 $O(h^2)$ 和 $O(h^4)$ 阶。这说明当步长向 0 减少时，辛普生公式的误差比梯形公式的误差更快地收敛到 0。当 $f(x)$ 的导数已知时，可用公式：

$$E_T(f,h)=\frac{-(b-a)f^{(2)}(c)h^2}{12} \text{和} E_S(f,h)=\frac{-(b-a)f^{(4)}(c)h^4}{180}$$

来估计得到给定精度的近似所需要的子区间数。

推论 7.2(梯形公式的误差分析)　设区间 $[a,b]$ 被分为宽度为 $h=(b-a)/M$ 的 M 个子区间 $[x_k,x_{k+1}]$，组合梯形公式：

$$T(f,h)=\frac{h}{2}(f(a)+f(b))+h\sum_{k=1}^{M-1}f(x_k) \tag{7}$$

是对积分：

$$\int_a^b f(x)dx = T(f,h) + E_T(f,h) \tag{8}$$

的一个逼近。若 $f\in C^2[a,b]$,则存在一个值 $c,a<c<b$,使得误差项 $E_T(f,h)$具有如下形式:

$$E_T(f,h)=\frac{-(b-a)f^{(2)}(c)h^2}{12}=O(h^2) \tag{9}$$

证明:首先确定公式在区间$[x_0,x_1]$上的误差项。对拉格朗日多项式 $P_1(x)$进行积分,其余项为:

$$\int_{x_0}^{x_1} f(x)dx = \int_{x_0}^{x_1} P_1(x)dx + \int_{x_0}^{x_1} \frac{(x-x_0)(x-x_1)f^{(2)}(c(x))}{2!}dx \tag{10}$$

在区间$(x-x_0)(x-x_1)$上,项$[x_0,x_1]$符号不变,而 $f^{(2)}(c(x))$连续,故积分的第二中值定理隐含说明,存在一个值 c_1,使得:

$$\int_{x_0}^{x_1} f(x)dx = \frac{h}{2}(f_0+f_1) + f^{(2)}(c_1)\int_{x_0}^{x_1} \frac{(x-x_0)(x-x_1)}{2!}dx \tag{11}$$

对(11)式右端的积分进行变量代换 $x=x_0+ht$:

$$\begin{aligned}\int_{x_0}^{x_1} f(x)dx &= \frac{h}{2}(f_0+f_1) + \frac{f^{(2)}(c_1)}{2}\int_0^1 h(t-0)h(t-1)hdt \\ &= \frac{h}{2}(f_0+f_1) + \frac{f^{(2)}(c_1)h^3}{2}\int_0^1 (t^2-t)dt \\ &= \frac{h}{2}(f_0+f_1) - \frac{f^{(2)}(c_1)h^3}{12}\end{aligned} \tag{12}$$

将所有区间$[x_k,x_{k+1}]$上的误差项相加,得:

$$\begin{aligned}\int_a^b f(x)dx &= \sum_{k=1}^{M}\int_{x_{k-1}}^{x_k} f(x)dx \\ &= \sum_{k=1}^{M}\frac{h}{2}(f(x_{k-1})+f(x_k)) - \frac{h^3}{12}\sum_{k=1}^{M} f^{(2)}(c_k)\end{aligned}$$

第一个求和式为组合梯形公式 $T(f,h)$,在第二项中,用 h 的一个等式 $h=(b-a)/M$ 进行替换,得:

$$\int_a^b f(x)dx = T(f,h) - \frac{(b-a)h^2}{12}\left(\frac{1}{M}\sum_{k=1}^{M} f^{(2)}(c_k)\right)$$

括号中的项可看作是 2 阶导数的一个均值,故可由 $f^{(2)}(c)$替换。从而得到:

$$\int_a^b f(x)dx = T(f,h) - \frac{(b-a)f^{(2)}(c)h^2}{12}$$

推论 7.2 得证。

推论 7.3 (辛普生公式的误差分析) 设$[a,b]$被分为宽度为 $h=(b-a)/(2M)$的 $2M$ 个等宽度子区间,组合辛普生公式:

$$S(f,h) = \frac{h}{3}(f(a)+f(b)) + \frac{2h}{3}\sum_{k=1}^{M-1} f(x_{2k}) + \frac{4h}{3}\sum_{k=1}^{M} f(x_{2k-1}) \tag{14}$$

为积分:

$$\int_a^b f(x)dx = S(f,h) + E_s(f,h) \tag{15}$$

的一个逼近。并且,若 $f\in C^4[a,b]$,则存在一个值 c,$a<c<b$,使得误差项 $E_s(f,h)$具有:

$$E_s(f,h)=\frac{-(b-a)f^{(4)}(c)h^4}{180}=O(h^4) \tag{16}$$

的形式。

例 7.7 考虑 $f(x)=2+\sin(2\sqrt{x})$,计算在区间[1,6]内使用组合梯形公式,且子区间数为 10、20、40、80 和 160 时的误差。

表 7.2 $f(x)=2+\sin(2\sqrt{x})$在[1,6]内的组合梯形公式。

M	h	$T(f,h)$	$E_T(f,h)=O(h^2)$
10	0.5	8.193 854 57	-0.010 375 40
20	0.25	8.186 049 26	-0.002 570 06
40	0.125	8.184 120 19	-0.000 640 98
80	0.0625	8.183 639 36	-0.000 160 15
160	0.03125	8.183 519 24	-0.000 040 03

表 7.2 给出了逼近 $T(f,h)$,$f(x)$的不定积分为:

$$F(x)=2x-\sqrt{x}\cos(2\sqrt{x})+\frac{\sin(2\sqrt{x})}{2}$$

而定积分的真值为:

$$\int_1^6 f(x)dx = F(x)\Big|_{x=1}^{x=6} = 8.18347920770$$

该值用来计算表 7.2 中的 $E_T(f,h)=8.1834792077-T(f,h)$。观察到当 h 减少 $E_T(f,h)$ 时,所得的误差减至约$\frac{1}{4}$,这一点很重要,因为它证实了误差的阶为 $O(h^2)$。

例 7.8 考虑 $f(x)=2+\sin(2\sqrt{x})$,计算在区间[1,6]内使用组合辛普生公式,且子区间数为 10、20、40、80 和 160 时的误差。

表 7.3 给出了逼近 $S(f,h)$,积分的真值为 8.1834792077,用它来计算表 7.3 中的 $E_S(f,h)=8.1834792077-S(f,h)$,注意到当 h 减少$\frac{2}{h}$时,相应的误差 $E_S(f,h)$减少为原来的约$\frac{1}{16}$,这证实了误差的阶为 $O(h^4)$。

例 7.9 计算 M 和步长 h,使得组合梯形公式对逼近 $E_T(f,h)$的误差 5×10^{-9}小于 $\int_2^7 dx/x\approx T(f,h)$。

被积函数为 $f(x)=1/x$,其前 2 阶导数为 $f'(x)=-1/x^2$ 和 $f^{(2)}(x)=2/x^3$。[2,7]内 $|f^{(2)}(x)|$的最大值在端点 $x=2$ 处取得,故对$|f^{(2)}(c)|\leqslant|f^{(2)}(2)|=\frac{1}{4}$,有 $2\leqslant c\leqslant7$。代入式(9),得:

$$|E_T(f,h)|=\frac{|-(b-a)f^{(2)}(c)h^2|}{12}\leqslant\frac{(7-2)\frac{1}{4}h^2}{12}=\frac{5h^2}{48} \tag{17}$$

表 7.3 区间[1,6]内 $f(x)=2+\sin(2\sqrt{x})$ 的组合辛普生公式

M	h	$S(f,h)$	$E_s(f,h)=O(h^4)$
5	0.5	8.18301549	0.00046371
10	0.25	8.18344750	0.00003171
20	0.125	8.18347717	0.00000204
40	0.0625	8.18347908	0.00000013
80	0.03125	8.18347920	0.00000001

步长 h 和 M 满足关系 $h=5/M$,代入式(17),得关系式:

$$|E_T(f,h)|\leqslant\frac{125}{48M^2}\leqslant 5\times10^{-9} \tag{18}$$

重写式(18),使之易于求解 M:

$$\frac{25}{48}\times10^9\leqslant M^2 \tag{19}$$

对式(19)求解,得 $22821.77\leqslant M$。由于 M 必须为整数,选择 $M=22\,822$,对应的步长为 $h=5/22\,822=0.000219086846$。当用这么多次函数求值来计算组合梯形公式的值时,很有可能函数的舍入误差会相当大。用该值进行计算,结果为:

$$T\left(f,\frac{5}{22\,822}\right)=1.252762969$$

与真值 $\int_2^7 dx/x=\ln(x)|_{x=2}^{x=7}=1.252762968$ 相比,误差较预测的小,因为使用了 $|f^{(2)}(c)|$ 的 $\frac{1}{4}$ 界。实验表明,要达到精度 5×10^{-9},需要 10 001 次函数求值,当使用 $M=10\,000$ 进行计算时,结果为:

$$T\left(f,\frac{5}{10\,000}\right)=1.252762973$$

组合梯形公式通常要求大量的函数求值才能得到准确的答案。与下面例子中的辛普生公式相比,后者只需少量的函数求值。

例 7.10 计算 M 和步长 h,使得组合辛普生公式的误差 $E_S(f,h)$ 比逼近式 $\int_2^7 dx/x\approx S(f,h)$ 的误差 5×10^{-9} 小。

被积函数为 $f(x)=1/x$,而 $f^{(4)}(x)=24/x^5$。[2,7]内在端点 $x=2$ 处可取得 $|f^{(4)}(c)|$ 的最大值,从而得到,对 $2\leqslant c\leqslant7$,有 $|f^{(4)}(c)|\leqslant|f^{(4)}(2)|=\frac{3}{4}$。代入式(16),得:

$$|E_S(f,h)|=\frac{|-(b-a)f^{(4)}(c)h^4|}{180}\leqslant\frac{(7-2)\frac{3}{4}h^4}{180}=\frac{h^4}{48} \tag{20}$$

步长 h 和 M 满足关系 $h=5/(2M)$,代入式(20),得关系式:

$$|E_S(f,h)|\leqslant\frac{625}{768M^4}\leqslant5\times10^{-9} \tag{21}$$

重写式(21),使之易于求解 M:

$$\frac{125}{768}\times10^9\leqslant M^4 \tag{22}$$

对式(22)求解,得 $112.95 \le M$。由于 M 必须为整数,选择 $M = 113$,对应的步长为 $h = 5/226 = 0.02212389381$。当计算组合辛普生公式时,结果为:

$$S\left(f, \frac{5}{226}\right) = 1.252762969$$

与 $\int_2^7 dx/x = \ln(x) \mid_{x=2}^{x=7} = 1.252762968$ 一致。实验表明,得到精度 5×10^{-9} 需要大约 129 次函数求值。当用 $M = 64$ 进行计算时,结果为:

$$S\left(f, \frac{5}{128}\right) = 1.252762973$$

于是可知,使用 229 次 $f(x)$ 求值的组合辛普生公式与使用 22 823 次 $f(x)$ 求值的组合梯形公式得到同样的精度。在例 7.10 中,辛普生公式的函数求值次数只有梯形公式的 $\frac{1}{100}$。

程序 7.1(组合梯形公式) 通过 $f(x)$ 的 $M + 1$ 个等步长采样点:

$$\int_a^b f(x)\,dx \approx \frac{h}{2}(f(a) + f(b)) + h\sum_{k=1}^{M-1} f(x_k)\ ,k = 0,1,2,\cdots,M$$

逼近积分 $x_k = a + kh$。注意:$x_0 = a$,而 $x_M = b$

```
function s = traprl(f,a,b,M)
% Input     - f is the integrand input as a string 'f'
%           - a and b are upper and lower limits of integration
%           - M is the number of subintervals
% Output    - s is the trapezoidal rule sum
h = (b - a)/M;
s = 0;
for k = 1:(M - 1)
   x = a + h * k;
   s = s + feval(f,x);
end
s = h * (feval(f,a) + feval(f,b))/2 + h * s;
```

程序 7.2(组合辛普生公式) 通过 f(x)的 2M + 1 个等步长采样点:

$$\int_a^b f(x)\,dx \approx \frac{h}{3}(f(a) + f(b)) + \frac{2h}{3}\sum_{k=1}^{M-1} f(x_{2k}) + \frac{4h}{3}\sum_{k=1}^{M} f(x_{2k-1})\ ,k = 0,1,2,\cdots,2M$$

逼近积分 $x_k = a + kh$。注意:$x_0 = a$,而 $x_{2M} = b$

```
function s = simprl(f,a,b,M)
% Input  - f is the integrand input as a string 'f'
%        - a and b are upper and lower limits of integration
%        - M is the number of subintervals
% Output - s is the simpson rule sum
h = (b - a)/(2 * M);
s1 = 0;
s2 = 0;
for k = 1:M
  x = a + h * (2 * k - 1);
```

```
    s1 = s1 + feval(f,x)
  end
  for k = 1:(M-1)
    x = a + h * 2 * k;
    s2 = s2 + feval(f,x);
  end
  s = h * (feval(f,a) + feval(f,b) + 4 * s1 + 2 * s2)/3;
```

7.2.2 习题

1. (i) 用 $M = 10$ 的组合梯形公式求下列每个积分。

   (ii) 用 $M = 5$ 的组合辛普生公式求下列每个积分。

   (a) $\int_{-1}^{1}(1 + x^2)^{-1}dx$

   (b) $\int_{0}^{1}(2 + \sin(2\sqrt{x}))dx$

   (c) $\int_{0.25}^{4}dx/\sqrt{x}$

   (d) $\int_{0}^{4}x^2e^{-x}dx$

   (e) $\int_{0}^{2}2x\cos(x)dx$

   (f) $\int_{0}^{\pi}\sin(2x)e^{-x}dx$

2. 曲线长。曲线 $y = f(x)$在区间 $a \leqslant x \leqslant b$ 的弧长为:

$$\text{弧长} = \int_a^b \sqrt{1 + (f'(x)^2)}\,dx$$

   (i) 用 $M = 10$ 的组合梯形公式求下列每个函数的弧长。

   (ii) 用 $M = 5$ 的组合辛普生公式求下列每个函数的弧长。

   (a) $f(x) = x^3$, $0 \leqslant x \leqslant 1$

   (b) $f(x) = \sin(x)$, $0 \leqslant x \leqslant \pi/4$

   (c) $f(x) = e^{-x}$, $0 \leqslant x \leqslant 1$

3. 表面积。由曲线 $y = f(x), a \leqslant x \leqslant b$,绕 x 轴旋转得到的立体,其表面积根据

$$\text{面积} = 2\pi\int_a^b f(x)\sqrt{1 + (f'(x))^2}\,dx \quad \text{计算:}$$

   (i) 用 $M = 10$ 的组合梯形公式求下列每个表面积。

   (ii) 用 $M = 5$ 的组合辛普生公式求下列每个表面积。

   (a) $f(x) = x^3$, $0 \leqslant x \leqslant 1$

   (b) $f(x) = \sin(x)$, $0 \leqslant x \leqslant \pi/4$

   (c) $f(x) = e^{-x}$, $0 \leqslant x \leqslant 1$

4. (a) 证明:梯形公式 ($M = 1, h = 1$) 对区间[0,1]内形如 $f(x) = c_1x + c_0$ 的次数小于等于 1 的多项式是精确的。

   (b) 利用被积函数 $f(x) = c_2x^2$,证明:梯形公式 ($M = 1, h = 1$) 在区间[0,1]内有误差项:

$$E_T(f,h)=\frac{-(b-a)f^{(2)}(c)h^2}{12}$$

5. (a) 证明:辛普生公式 ($M=1,h=1$) 对区间[0,2]内形如 $f(x)=c_3x^3+c_2x^2+c_1x+c_0$ 的次数小于等于 3 的多项式是精确的。

   (b) 利用被积函数 $f(x)=c_4x^4$,证明:辛普生公式 ($M=1,h=1$) 在区间[0,2]内有误差项:

$$E_S(f,h)=\frac{-(b-a)f^{(4)}(c)h^4}{180}$$

6. 用待定系数法推导出梯形公式($M=1,h=1$)。

   即:

   (a) 求常数 ω_0 和 ω_1,使得 $\int_0^1 g(t)\,dt=\omega_0g(0)+\omega_1g(1)$ 对函数 $g(t)=1$ 和 $g(t)=t$ 是精确的。

   (b) 利用关系式 $f(x_0+ht)=g(t)$和变量替换 $x=x_0+ht$ 和 $dx=hdt$,将梯形公式由区间[0,1]平移到区间$[x_0,x_1]$。

   (a) 的提示:可以得到关于 2 个未知量 ω_0 和 ω_1 的线性方程组。

7. 用待定系数法推导出辛普生公式 ($M=1,h=1$)。

   (a) 求常数 ω_0,ω_1 和 ω_2,使得 $\int_0^2 g(t)=\omega_0g(0)+\omega_1g(1)+\omega_2g(2)$ 对函数 $g(t)=1,g(t)=t$ 和$g(t)=t^2$ 是精确的。

   (b) 利用关系式 $f(x_0+ht)=g(t)$和变量替换 $x=x_0+ht$ 和 $dx=hdt$,将梯形公式由区间[0,2]平移到区间$[x_0,x_2]$。

   (a) 的提示:可以得到关于 3 个未知量 ω_0,ω_1 和 ω_2 的线性方程组。

8. 计算 M 和步长 h,使得 M 个子区间的组合梯形公式可以用来计算以下的函数,且有精度 5×10^{-9}:

   (a) $\int_{-\pi/6}^{\pi/6}\cos(x)\,dx$

   (b) $\int_2^3\frac{1}{5-x}dx$

   (c) $\int_0^2 xe^{-x}dx$

   (c) 的提示:$f^{(2)}(x)=(x-2)e^{-x}$

9. 计算 M 和步长 h,使得 $2M$ 个子区间的组合辛普生公式可以用来计算以下的函数,且有精度 5×10^{-9}:

   (a) $\int_{-\pi/6}^{\pi/6}\cos(x)\,dx$

   (b) $\int_2^3\frac{1}{5-x}dx$

   (c) $\int_0^2 xe^{-x}dx$

   (c) 的提示:$f^{(4)}(x)=(x-4)e^{-x}$

10. 考虑定积分 $\int_{-0.1}^{0.1}\cos(x)\,dx=2\sin(0.1)=0.1996668333$,下表给出了组合梯形公式的

近似值,计算 $E_T(f,h)=0.199668-T(f,h)$,并证实其精度为 $O(h^2)$。下表给定为:

M	h	$S(f,h)$	$E_T(f,h)=O(h^2)$
1	0.2	0.199 000 8	
2	0.1	0.199 500 4	
4	0.05	0.199 625 2	
8	0.025	0.199 656 4	
16	0.0125	0.199 664 2	

11. 考虑定积分 $\int_{-0.75}^{0.75}\cos(x)\,dx = 2\sin(0.75) = 1.363277520$,下表给出了组合辛普生公式的近似值,计算 $E_S(f,h)=1.3632775-S(f,h)$,并证实其精度为 $O(h^4)$。下表给定为:

M	h	$S(f,h)$	$E_T(f,h)=O(h^2)$
1	0.75	1.365 844 4	
2	0.375	1.363 429 8	
4	0.187 5	1.363 286 9	
8	0.093 75	1.363 278 1	

12. 中点公式。$[x_0,x_1]$的中点公式为:

$$\int_{x_0}^{x_1}f(x)\,dx = hf\left(x_0+\frac{h}{2}\right)+\frac{h^3}{24}f^{(2)}(c_1)\text{，其中 } h=\frac{x_1-x_0}{2}$$

则有:

(a) 将 $f(x)$的不定积分 $F(x)$在点 $x_0+h/2$ 展开为泰勒级数,并建立$[x_0,x_1]$上的中点公式。

(b) 用(a)的结果,证明:$[a,b]$内 $f(x)$积分的组合中点公式为:

$$M(f,h) = h\sum_{k=1}^{N}f\left(a+\left(k-\frac{1}{2}\right)h\right) \quad , \quad h=\frac{b-a}{N}$$

这是 $f(x)$在$[a,b]$内的积分的一种逼近,写为:

$$\int_a^b f(x)\,dx \approx M(f,h)$$

(c) 证明:(b)的误差项 $E_M(f,h)$为:

$$E_M(f,h) = \frac{h^3}{24}\sum_{k=1}^{N}f^{(2)}(c_k) = \frac{(b-a)f^{(2)}(c)h^2}{24} = O(h^2)$$

13. 用 $M=10$ 的中点公式计算习题 1 中的积分。

14. 证明推论 7.3。

7.2.3 算法与程序

1. (a) 对习题 1 中的每项计算 M 和步长 h,使得可以用组合梯形公式计算给定的积分,并精确到小数点后第 9 位。用程序 7.1 求每个积分。

   (b) 对习题 1 中的每项,计算 M 和步长 h,使得可以用组合辛普生公式计算给定的积分,并精确到小数点后第 9 位。用程序 7.2 求每个积分。

2. 用程序 7.2 求习题 2 中的定积分,精确到小数点后第 11 位:

3. 组合梯形公式可用于求只有若干点函数值已知的函数积分。将程序7.1改写为求区间$[a,b]$内过M个给定点的函数$f(x)$的积分逼近(注意:节点不需要是等距的)。利用该程序求过点$\{(\sqrt{k^2+1},k^{1/3})\}_{k=0}^{13}$的函数的积分逼近。
4. 组合辛普生公式可用于求只有若干点函数值已知的函数积分。将程序7.2改写为求区间$[a,b]$内过M个给定点的函数$f(x)$的积分逼近(注意:节点不需要是等距的)。利用该程序求过点$\{(\sqrt{k^2+1},k^{1/3})\}_{k=0}^{13}$的函数的积分逼近。
5. 修改程序7.1,使之用组合中点公式(习题12)来逼近函数$f(x)$在$[a,b]$内积分。利用该程序求习题1中的定积分,精确到小数点后第11位。
6. 使用本节的任意算法,求下面每个定积分的逼近,精确到小数点后第10位:

   (a) $\int_{1/7\pi}^{1/4\pi}\sin(1/x)\,dx$

   (b) $\int_{\frac{1}{5\pi}+10^{-5}}^{\frac{1}{4\pi}-10^{-5}}\frac{1}{\sin(1/x)}\,dx$
7. 下面的例子说明如何用辛普生公式来求积分方程的近似解。用辛普生公式和$h=1/2$来求解方程$v(x)=x^2+0.1\int_0^1(x^2+t)v(t)dt$,设$t_0=0,t_1=1/2$和$t_2=1$,则:

$$\int_0^1(x^2+t)v(t)dt\approx\frac{1/2}{3}((x_n^2+0)v_0+4(x_n^2+\frac{1}{2})v_1+(x_n^2+1)v_2)$$

令:

$$v(x_n)=x_n^2+0.1\left(\frac{1}{6}((x_n^2+0)v_0+4(x_n^2+\frac{1}{2})v_1+(x_n^2+1)v_2)\right)\tag{1}$$

将$x_0=0,x_1=1/2$和$x_2=1$代入式(1),得线性方程组:

$$\begin{aligned}v_0&=0+\frac{1}{60}((0)v_0+2v_1+v_2)\\v_1&=\frac{1}{4}+\frac{1}{60}\left(\frac{1}{4}v_0+3v_1+\frac{5}{4}v_2\right)\\v_2&=1+\frac{1}{60}(v_0+6v_1+2v_2)\end{aligned}\tag{2}$$

将式(2)的解($v_0=0.0273,v_1=0.2866,v_2=1.0646$)代入式(1)并简化之,得:

$$v(x)\approx1.037305x^2+0.027297\tag{3}$$

(a) 作为验证,将该解代入到积分方程的右端,对其进行积分,并与式(3)的结果比较。

(b) 利用组合辛普生公式和$h=0.5$来求积分方程:

$$v(x)=x^2+0.1\int_0^1(x^2+t)v(t)dt$$

的近似解。并用(a)中的过程验证该解。

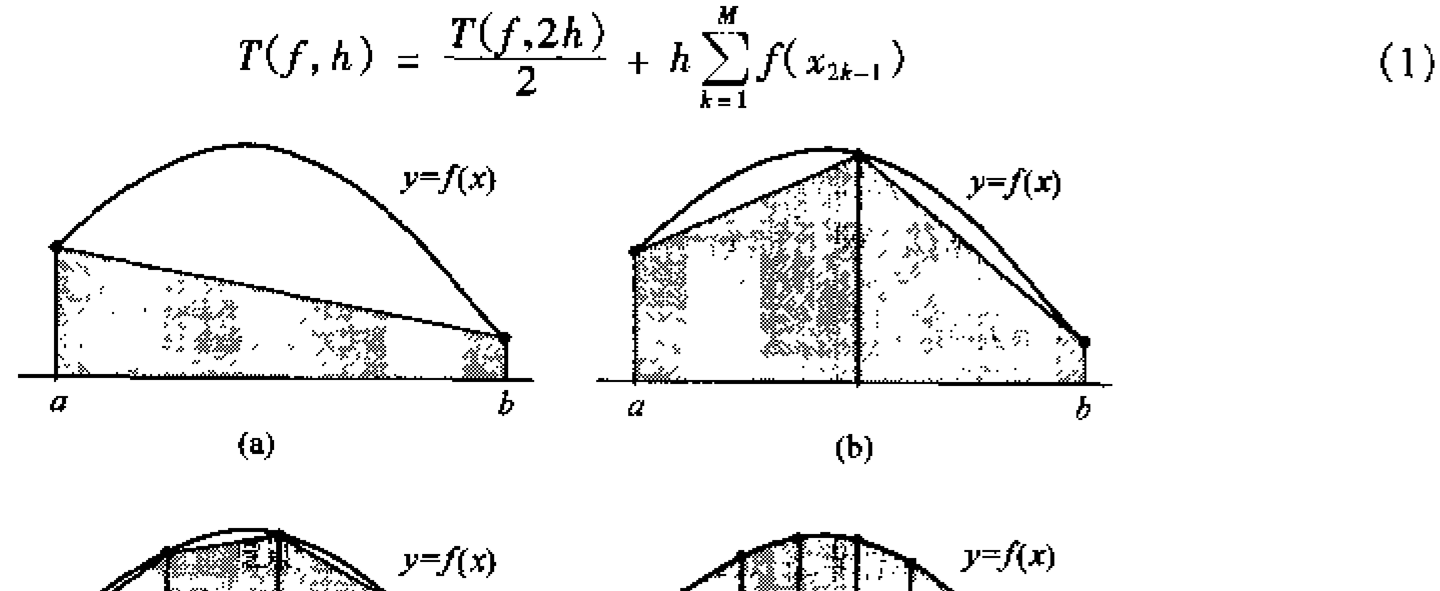
7.3 递归公式与龙贝格积分

本节中我们将说明如何用梯形公式的线性组合计算辛普生逼近。如果用大量的子区间,则该逼近会有很高的精度,如何选择子区间的数目?下面的过程将通过对2个子区间,4个子

区间,...,进行试验,直至得到想要的精度来回答这一问题。首先要生成一个梯形逼近的序列$\{T(J)\}$,当子区间数目增加一倍时,函数求值的次数也近似加倍,因为必须在所有的先前的点和先前区间的中点对函数进行求值(见图 7.8)。

定理 7.4 (连续梯形公式) 设 $J\geqslant 1$,点$\{x_k = a + kh\}$将$[a,b]$划分为 $2^J = 2M$ 个宽度为 $h = (b-a)/2^J$ 的子区间。梯形公式 $T(f,h)$和 $T(f,2h)$满足如下关系:

$$T(f,h) = \frac{T(f,2h)}{2} + h\sum_{k=1}^{M} f(x_{2k-1}) \tag{1}$$



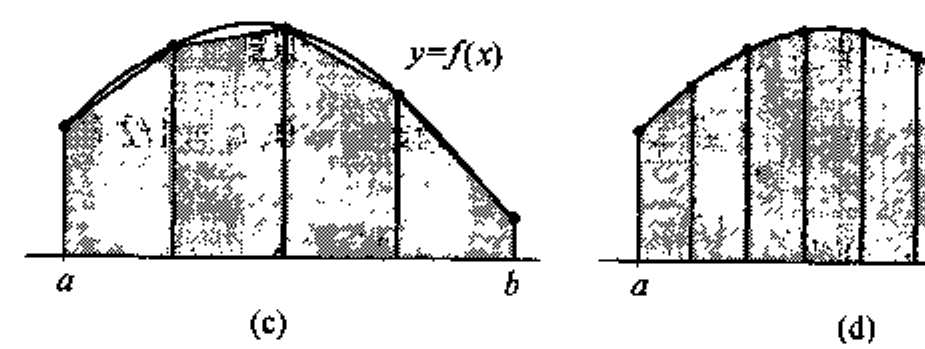


图 7.8 (a) $T(0)$为 $2^0 = 1$ 个梯形的面积
(b) $T(1)$为 $2^1 = 2$ 个梯形的面积
(c) $T(2)$为 $2^2 = 4$ 个梯形的面积
(d) $T(3)$为 $2^3 = 8$ 个梯形的面积

定义 7.3 (梯形公式序列) 定义 $T(0) = \frac{h}{2}(f(a)+f(b))$为步长为 $h = b-a$ 的梯形公式,则对 $J\geqslant 1$,定义 $T(J) = T(f,h)$,其中 $T(f,h)$是步长为$(b-a)/2^J$ 的梯形公式。

推论 7.4 (递归梯形公式) 由 $T(0) = \frac{h}{2}(f(a)+f(b))$开始,由如下递归公式可生成一个梯形公式$\{T(J)\}$的序列:

$$T(J) = \frac{T(J-1)}{2} + h\sum_{k=1}^{M} f(x_{2k-1}), \quad J = 1,2,\cdots \tag{2}$$

其中 $h = (b-a)/2^J$,并且$\{x_k = a + kh\}$。

证明:对偶节点 $x_0 < x_2 < \cdots < x_{2M-2} < x_{2M}$,使用步长为 $2h$ 的梯形公式:

$$T(J-1) = \frac{2h}{2}(f_0 + 2f_2 + 2f_4 + \cdots + 2f_{2M-4} + 2f_{2M-2} + f_{2M}) \tag{3}$$

对所有节点 $x_0 < x_1 < x_2 < \cdots < x_{2M-1} < x_{2M}$,使用步长为 h 的梯形公式:

$$T(J) = \frac{h}{2}(f_0 + 2f_1 + 2f_2 + \cdots + 2f_{2M-2} + 2f_{2M-1} + f_{2M}) \tag{4}$$

收集式(4)中下标为奇数和偶数的项,得:

$$T(J) = \frac{h}{2}(f_0 + 2f_2 + \cdots + 2f_{2M-2} + f_{2M}) + h\sum_{k=1}^{M} f_{2k-1} \tag{5}$$

将式(3)代入式(5),得 $T(J) = T(J-1)/2 + h\sum_{k=1}^{M} f_{2k-1}$,证毕。

例 7.11　用连续梯形公式计算积分 $\int_1^5 dx/x = \ln(5) - \ln(1) = 1.609437912$ 的逼近式 $T(0)$, $T(1)$, $T(2)$ 和 $T(3)$。

表 7.4 给出了计算 $T(3)$ 所需的 9 个值和计算 $T(1)$, $T(2)$ 和 $T(3)$ 所需的中点值。求值的详细过程如下:

当 $h=4$: $T(0) = \frac{4}{2}(1.000000 + 0.200000) = 2.400000$

当 $h=2$: $T(1) = \frac{T(0)}{2} + 2(0.333333)$

$= 1.200000 + 0.666666 = 1.866666$

当 $h=1$: $T(2) = \frac{T(1)}{2} + 1(0.500000 + 0.250000)$

$= 0.933333 + 0.750000 = 1.683333$

当 $h=\frac{1}{2}$: $T(3) = \frac{T(2)}{2} + \frac{1}{2}(0.666667 + 0.400000$

$+ 0.285714 + 0.222222)$

$= 0.841667 + 0.787302 = 1.628968$

下面的结果说明了梯形公式与辛普生公式之间的重要关系。用步长 $2h$ 和 h 来计算梯形公式的结果分别为 $T(f,2h)$ 和 $T(f,h)$。用这些值的组合可得辛普生公式:

$$S(f,h) = \frac{4T(f,h) - T(f,2h)}{3} \tag{6}$$

定理 7.5 (递归辛普生公式)　设 $\{T(J)\}$ 为由推论 7.4 产生的梯形公式序列,若 $J \geq 1$,且 $S(J)$ 为区间 $[a,b]$ 的 2^J 个辛普生公式,则 $S(J)$ 和 $T(J-1)$, $T(J)$ 满足关系式:

$$S(J) = \frac{4T(J) - T(J-1)}{3}, \quad J = 1,2,\cdots \tag{7}$$

表 7.4　用来计算 $T(3)$ 的 9 个点和计算 $T(1)$, $T(2)$ 和 $T(3)$ 需要的中点值

x	$f(x)=\frac{1}{x}$	计算 $T(0)$ 需要的端点值	计算 $T(1)$ 需要的中点值	计算 $T(2)$ 需要的中点值	计算 $T(3)$ 需要的中点值
1.0	1.000000	1.000000			
1.5	0.666667				0.666667
2.0	0.500000			0.500000	
2.5	0.400000				0.400000
3.0	0.333333		0.333333		
3.5	0.285714				0.285714
4.0	0.250000			0.250000	
4.5	0.222222				0.222222
5.0	0.200000	0.200000			

证明:由步长为 h 的梯形公式 $T(J)$得到逼近:

$$\int_a^b f(x)\,dx \approx \frac{h}{2}(f_0 + 2f_1 + 2f_2 + \cdots + 2f_{2M-2} + 2f_{2M-1} + f_{2M})$$
$$= T(J) \tag{8}$$

由步长为 $2h$ 的梯形公式 $T(J-1)$得到逼近:

$$\int_a^b f(x)\,dx \approx h(f_0 + 2f_2 + \cdots + 2f_{2M-2} + f_{2M}) = T(J-1) \tag{9}$$

将(8)式乘以 4,得:

$$4\int_a^b f(x)\,dx \approx h(2f_0 + 4f_1 + 4f_2 + \cdots + 4f_{2M-2} + 4f_{2M-1} + 2f_M)$$
$$= 4T(J) \tag{10}$$

式(10)减去式(9)得:

$$3\int_a^b f(x)\,dx \approx h(f_0 + 4f_1 + 2f_2 + \cdots + 2f_{2M-2} + 4f_{2M-1} + f_{2M})$$
$$= 4T(J) - T(J-1) \tag{11}$$

该式重写为:

$$\int_a^b f(x)\,dx \approx \frac{h}{3}(f_0 + 4f_1 + 2f_2 + \cdots + 2f_{2M-2} + 4f_{2M-1} + f_{2M})$$
$$= \frac{4T(J) - T(J-1)}{3}$$

式(12)中的中项为辛普生公式 $S(J) = S(f,h)$,从而定理得证。

例 7.12 用连续辛普生公式求例 7.11 中的积分逼近式 $S(1)$,$S(2)$和 $S(3)$。

利用例 7.11 中的结果和公式(7)及 $J = 1,2,3$,计算得:

$$S(1) = \frac{4T(1) - T(0)}{3} = \frac{4(1.866666) - 2.400000}{3} = 1.688888$$

$$S(2) = \frac{4T(2) - T(1)}{3} = \frac{4(1.683333) - 1.866666}{3} = 1.622222$$

$$S(3) = \frac{4T(3) - T(2)}{3} = \frac{4(1.628968) - 1.683333}{3} = 1.610846$$

在 7.1 节中,布尔公式由定理 7.1 给出,它是通过对基于节点 x_0, x_1, x_2, x_3 和 x_4 的 4 次拉格朗日多项式求积分得到的。另一种建立布尔公式的方法在习题中给出。当对区间$[a,b]$内宽度为 $h = (b-a)/(4M)$的 $4M$ 个等间距子区间应用 M 次布尔公式时,称之为组合布尔公式:

$$B(f,h) = \frac{2h}{45}\sum_{k=1}^{M}(7f_{4k-4} + 32f_{4k-3} + 12f_{4k-2} + 32f_{4k-1} + 7f_{4k})$$

下面的结果给出了连续布尔公式和辛普生公式的关系。

定理 7.6 (递归布尔公式) 设$\{S(J)\}$为由定理 7.5 产生的辛普生公式序列,若 $J \geq 2$ 且 $B(J)$为区间$[a,b]$内 2^J 个子区间的布尔公式,则 $B(J)$与辛普生公式 $S(J-1)$和 $S(J)$ 满足关系:

$$B(J) = \frac{16S(J) - S(J-1)}{15}, \quad J = 2,3,\cdots \tag{14}$$

证明留作练习。

例 7.13 用连续布尔公式求例 7.11 中积分的逼近 $B(2)$和 $B(3)$。

根据例 7.12 中的结果、式(14)及 $J=2$ 和 3,计算得:

$$B(2)=\frac{16S(2)-S(1)}{15}=\frac{16(1.622222)-1.688888}{15}=1.617778$$

$$B(3)=\frac{16S(3)-S(2)}{15}=\frac{16(1.610846)-1.622222}{15}=1.610088$$

读者可能对我们的目标感到疑惑,现在我们来证明式(7)和式(14)都是龙贝格积分的特例。对例 7.11 积分的下一级逼近为:

$$\frac{64B(3)-B(2)}{63}=\frac{64(1.610088)-1.617778}{63}=1.609490$$

该答案精确到小数点后第5位。

7.3.1 龙贝格积分

在 7.2 节中,我们知道组合梯形公式和组合辛普生公式的误差项 $E_T(f,h)$和 $E_s(f,h)$的阶数分别为 $O(h^2)$和 $O(h^4)$。不难证明,组合布尔公式的误差项 $E_B(f,h)$阶数为 $O(h^6)$,故有:

$$\int_a^b f(x)\,dx = T(f,h)+O(h^2) \tag{15}$$

$$\int_a^b f(x)\,dx = S(f,h)+O(h^4) \tag{16}$$

$$\int_a^b f(x)\,dx = B(f,h)+O(h^6) \tag{17}$$

余项(15)~(17)的意义如下:设一个逼近公式使用了步长 h 和 $2h$,然后对两个结果进行代数运算,得到改进的答案。每个改进将误差项的阶由 $O(h^{2N})$提高到 $O(h^{2N+2})$。该过程称为龙贝格积分,它有自己的优点和缺点。

与布尔公式相比,牛顿-柯蒂斯公式用得较少,这是因为9点牛顿-柯蒂斯面积公式中有负的权值,而超过10点的所有公式中都有负的权,这会导致由舍入带来的误差。龙贝格积分的优点在于其所有的权都是正的,且其等距的节点横坐标易于计算。

龙贝格积分的缺点之一是,为了将误差由 $O(h^{2N})$降低到 $O(h^{2N+2})$,函数求值次数增加了一倍。使用连续公式能减少计算量。龙贝格积分基于理论假设,若对所有的 N,有 $f\in C^N[a,b]$,则梯形公式的误差项可以表示为一个只包含 h 的偶数次幂的级数,即:

$$\int_a^b f(x)\,dx = T(f,h)+E_T(f,h) \tag{18}$$

其中:

$$E_T(f,h)=a_1h^2+a_2h^4+a_3h^6+\cdots \tag{19}$$

对公式(19)的推导可在参考文献[153]中找到。

由于公式(19)中只包含 h 的偶数次项,可以连续地使用理查逊改进(Richardson improvement),首先消去 a_1,接着消去 a_2,然后是 a_3,以此类推。该过程产生偶数阶次的误差项

$O(h^4)$,$O(h^6)$和 $O(h^8)$。我们将证明,第一次改进为 $2M$ 个区间的辛普生公式。由 $T(f,2h)$ 和 $T(f,h)$开始,有:

$$\int_a^b f(x)dx = T(f,2h) + a_1 4h^2 + a_2 16h^4 + a_3 64h^6 + \cdots \tag{20}$$

和:

$$\int_a^b f(x)dx = T(f,h) + a_1 h^2 + a_2 h^4 + a_3 h^6 + \cdots \tag{21}$$

将式(21)乘以 4,得:

$$4\int_a^b f(x)dx = 4T(f,h) + a_1 4h^2 + a_2 4h^4 + a_3 4h^6 + \cdots \tag{22}$$

通过用式(22)减式(20)消去 a_1,结果为:

$$3\int_a^b f(x)dx = 4T(f,h) - T(f,2h) - a_2 12h^4 - a_3 60h^6 - \cdots \tag{23}$$

用(23)式除以 3,并对其中的系数重新命名,得:

$$\int_a^b f(x)dx = \frac{4T(f,h) - T(f,2h)}{3} + b_1 h^4 + b_2 h^6 + \cdots \tag{24}$$

与(6)式相同,式(24)的右端第一个量为辛普生公式 $S(f,h)$。这说明 $E_s(f,h)$只包含 h 的偶数幂次项:

$$\int_a^b f(x)dx = S(f,h) + b_1 h^4 + b_2 h^6 + b_3 h^8 + \cdots \tag{25}$$

这了证明第二次改进为布尔公式,由式(25)开始,写出包含 $S(f,2h)$的公式:

$$\int_a^b f(x)dx = S(f,2h) + b_1 16h^4 + b_2 64h^6 + b_3 256h^8 + \cdots \tag{26}$$

当从式(25)和式(26)中消去 b_1 时,得到包含布尔公式的结果:

$$\begin{aligned}\int_a^b f(x)dx &= \frac{16S(f,h) - S(f,2h)}{15} - \frac{b_2 48h^6}{15} - \frac{b_3 240h^8}{15} - \cdots \\ &= B(f,h) - \frac{b_2 48h^6}{15} - \frac{b_3 240h^8}{15} - \cdots\end{aligned} \tag{27}$$

龙贝格积分的一般形式基于引理 7.1。

引理 7.1 (龙贝格积分的理查逊改进) 给定两个 Q 的逼近 $R(2h,K-1)$和 $R(h,K-1)$,满足:

$$Q = R(h,K-1) + c_1 h^{2K} + c_2 h^{2K+2} + \cdots \tag{28}$$

和:

$$Q = R(2h,K-1) + c_1 4^K h^{2K} + c_2 4^{K+1} h^{2k+2} + \cdots \tag{29}$$

有改进的逼近,形如:

$$Q = \frac{4^K R(h,K-1) - R(2h,K-1)}{4^K - 1} + O(h^{2K+2}) \tag{30}$$

证明留作练习。

定义 7.4 定义$[a,b]$内 $f(x)$的面积公式序列$\{R(J,K): J \geq K\}_{J=0}^{\infty}$如下:

$$\begin{aligned} R(J,0) &= T(J) \quad, \quad J\geqslant 0(\text{是连续梯形公式}) \\ R(J,1) &= S(J) \quad, \quad J\geqslant 1(\text{是连续辛普生公式}) \\ R(J,2) &= B(J) \quad, \quad J\geqslant 2(\text{是连续布尔公式}) \end{aligned} \tag{31}$$

第一个公式$\{R(J,0)\}$用来产生第一次改进$\{R(J,1)\}$，后者又用来产生第二次改进$\{R(J,2)\}$。我们已知道形式：

$$\begin{aligned} R(J,1) &= \frac{4^1 R(J,0) - R(J-1,0)}{4^1-1} \quad, \quad J\geqslant 1 \\ R(J,2) &= \frac{4^2 R(J,1) - R(J-1,1)}{4^2-1} \quad, \quad J\geqslant 2 \end{aligned} \tag{32}$$

在式(24)和式(27)中用式(31)中的符号来表示。构造改进的一般公式为：

$$R(J,K) = \frac{4^K R(J,K-1) - R(J-1,K-1)}{4^K-1} \quad, \quad J\geqslant K \tag{33}$$

为计算方便，值$R(J,K)$以表7.5中的方式组织为龙贝格积分表。

表7.5 龙贝格积分表

J	R(J,0) 梯形公式	R(J,1) 辛普生公式	R(J,2) 布尔公式	R(J,3) 第三次改进	R(J,4) 第四次改进
0	$R(0,0)$				
1	$R(1,0)$	$R(1,1)$			
2	$R(2,0)$	$R(2,1)$	$R(2,2)$		
3	$R(3,0)$	$R(3,1)$	$R(3,2)$	$R(3,3)$	
4	$R(4,0)$	$R(4,1)$	$R(4,2)$	$R(4,3)$	$R(4,4)$

例7.14 利用龙贝格积分计算定积分的近似值：

$$\int_0^{\pi/2}(x^2+x+1)\cos(x)\,dx = -2+\frac{\pi}{2}+\frac{\pi^2}{4} = 2.038197427067\cdots$$

表7.6给出计算过程，每一列中的数都收敛到2.038197427067…，辛普生公式的列比梯形公式的列收敛速度快。在本例中，相邻的两列中右边的列的速度快于左边的列。

表7.6 例7.14的龙贝格积分表

J	R(J,0) 梯形公式	R(J,1) 辛普生公式	R(J,2) 布尔公式	R(J,3) 第三次改进
0	0.785398163397			
1	1.726812656758	2.040617487878		
2	1.960534166564	2.038441336499	2.038296259740	
3	2.018793948078	2.038213875249	2.038198711166	2.038197162776
4	2.033347341805	2.038198473047	2.038197446234	2.038197426156
5	2.036984954990	2.038197492719	2.038197427363	2.038197427064

若我们考查其误差项$E(J,K) = -2+\pi/2+\pi^2/4-R(J,K)$，则表7.6中龙贝格值的收敛性更明显。设区间宽度为$h=b-a$，且$f(x)$的更高阶导数在同一量级上，龙贝格表第K

列的误差以 $1/2^{2K+2}=1/4^{K+1}$ 的收缩比例逐行递减。误差 $E(J,0)$ 的收缩因子为 1/4,误差 $E(J,1)$ 的收缩因子为 1/16,以此类推。这可以通过考查表 7.7 中的 $\{E(J,K)\}$ 得到。

表 7.7　例 7.14 的龙贝格误差表

J	h	$E(J,0)=O(h^2)$	$E(J,1)=O(h^4)$	$E(J,2)=O(h^6)$	$E(J,3)=O(h^8)$
0	$b-a$	−1.252799263670			
1	$\frac{b-a}{2}$	−0.311384770309	0.002420060811		
2	$\frac{b-a}{4}$	−0.077663260503	0.000243909432	0.000098832673	
3	$\frac{b-a}{8}$	−0.019403478989	0.000016448182	0.000001284099	−0.000000264291
4	$\frac{b-a}{16}$	−0.004850085262	0.000001045980	0.000000019167	−0.000000000912
5	$\frac{b-a}{32}$	−0.001212472077	0.000000065651	0.000000000296	−0.000000000003

定理 7.7 (龙贝格积分的精度)　设 $f\in C^{2k+2}[a,b]$,则龙贝格逼近的截断误差由公式:

$$\int_a^b f(x)\,dx = R(J,K)+b_K h^{2K+2} f^{(2K+2)}(C_{J,K})$$
$$= R(J,K)+O(h^{2K+2}) \tag{34}$$

给出。其中,$h=(b-a)/2^J$,b_K 为依赖于 K 的常数,且 $C_{J,K}\in[a,b]$。见参考文献[153]第 126 页。

例 7.15　应用定理 7.7,并证明:

$$\int_0^2 10x^9\,dx = 1024 \equiv R(4,4)$$

证:

被积函数为 $f(x)=10x^9$,且 $f^{(10)}(x)=0$。故值 $K=4$ 可使误差项恒为 0,通过数值计算可得 $R(4,4)=1\,024$。

程序 7.3(递归梯形公式)　利用梯形公式和连续增加$[a,b]$的子区间数来逼近:

$$\int_a^b f(x)\,dx \approx \frac{h}{2}\sum_{k=1}^{2^J}(f(x_{k-1})+f(x_k))$$

第 J 次循环在 2^J+1 个等距点处对 $f(x)$ 采样。

```
function T = rctrap(f,a,b,n)
% Input    - f is the integrand input as a string 'f'
%          - a and b are upper and lower limits of integration
%          - n is the number of times for recursion
% Output   - T is the recursive trapezoidal rule list

M = 1;
h = b - a;
T = zeros(1,n+1);
T(1) = h * (feval(f,a) + feval(f,b))/2;
```

```
for j = 1;
  M = 2 * M;
  h = h/2;
  s = 0;
  for k = 1:M/2
    x = a + h * (2 * k - 1);
    s = s + feval(f,x);
  end
  T(j + 1) = T(j)/2 + h * s;
end
```

程序 7.4（龙贝格积分） 通过生成 $J \geqslant K$ 的逼近表 $R(J,K)$，并以 $R(J+1,J+1)$ 为最终解来逼近积分：

$$\int_a^b f(x)\,dx \approx R(J,J)$$

逼近 $R(J,K)$ 保存在一个特别的下三角矩阵中，第 0 列的元素 $R(J,0)$ 用基于 2^J 个 $[a,b]$ 子区间的连续梯形公式计算，然后利用龙贝格公式计算 $R(J,K)$。
第 J 行的元素为：

$$R(J,K) = R(J,K-1) + \frac{R(J,K-1) - R(J-1,K-1)}{4^K - 1}\quad,\quad 1 \leqslant K \leqslant J$$

当 $|R(J,J) - R(J+1,J+1)| < \text{tol}$ 时，程序在第 $(J+1)$ 行结束

```
function [R,quad,err,h] = romber(f,a,b,n,tol)
% Input   - f is the integrand input as a string 'f'
%         - a and b are upper and lower limits of integration.
%         - n is the maximum number of rows in the table
%         - tol is the tolerance
% Output  - R is the Romberg table
%         - quad is the quadrature value
%         - err is the error estimate
%         - h is the smallest step size used
M = 1;
h = b - a;
err = 1;
J = 0;
R = zeros(4,4);
R(1,1) = h * (feval(f,a) + feval(f,b))/2
while((err > tol)&(J < n))|(J < 4)
   J = J + 1;
   h = h/2;
   s = 0;
   for p = 1:M
     x = a + h * (2 * p - 1);
     s = s + feval(f,x);
   end
   R(J + 1,1) = R(J,1)/2 + h * s;
  M = 2 * M;
  for K = 1:J
```

```
    R(J+1,K+1) = R(J+1,K) + (R(J+1,K) - R(J,K)/(4^K-1);
  end
  err = abs(R(J,J) - R(J+1,K+1));
end
quad = R(J+1,J+1);
```

7.3.2 习题

1. 对下面每个定积分,构造(手算)一个 3 行的龙贝格表(表 7.5):

   (a) $\int_0^3 \frac{\sin(2x)}{1+x^2}dx = 0.6717578646\cdots$

   (b) $\int_0^3 \sin(4x)e^{-2x}dx = 0.1997146621\cdots$

   (c) $\int_{0.04}^1 \frac{1}{\sqrt{x}}dx = 1.6$

   (d) $\int_0^2 \frac{1}{x^2+\frac{1}{10}}dx = 4.4713993943\cdots$

   (e) $\int_{1/(2\pi)}^2 \sin\left(\frac{1}{x}\right)dx = 1.1140744942\cdots$

   (f) $\int_0^2 \sqrt{4-x^2}\,dx = \pi = 3.1415926535\cdots$

2. 设连续梯形公式收敛到 L(即 $\lim_{J\to\infty} T(J) = L$)。

   (a) 证明:连续辛普生公式收敛到 L(即 $\lim_{J\to\infty} S(J) = L$)。

   (b) 证明:连续布尔公式收敛到 L(即 $\lim_{J\to\infty} B(J) = L$)。

3. (a) 证明:布尔公式($M=1,h=1$)对[0,4]内形如 $f(x)=c_5x^5+c_4x^4+\cdots+c_1x+c_0$ 的小于等于 5 次的多项式是精确的。

   (b) 利用被积函数 $f(x)=c_6x^6$,证明:布尔公式($M=1,h=1$)在区间[0,4]内的误差项为:

$$E_B(f,h) = \frac{-2(b-a)f^{(6)}(c)h^6}{945}$$

4. 利用待定系数法推导布尔公式($M=1,h=1$):计算常数 $\omega_0,\omega_1,\omega_2,\omega_3$ 和 ω_4,使得:

$$\int_0^4 g(t)\,dt = \omega_0g(0)+\omega_1g(1)+\omega_2g(2)+\omega_3g(3)+\omega_4g(4)$$

   对 5 个函数 $g(t)=1,t,t^2,t^3$ 和 t^4 是精确的。提示:可得到线性方程组:

$$\begin{aligned}
\omega_0+\omega_1+\omega_2+\omega_3+\omega_4 &= 4\\
\omega_1+2\omega_2+3\omega_3+4\omega_4 &= 8\\
\omega_1+4\omega_2+9\omega_3+16\omega_4 &= \frac{64}{3}\\
\omega_1+8\omega_2+27\omega_3+64\omega_4 &= 64\\
\omega_1+16\omega_2+81\omega_3+256\omega_4 &= \frac{1024}{5}
\end{aligned}$$

5. 对 $J=2$ 的情况，建立关系 $B(J)=(16S(J)-S(J-1))/15$，利用如下信息：

$$S(1)=\frac{2h}{3}(f_0+4f_2+f_4)$$

和：

$$S(2)=\frac{h}{3}(f_0+4f_1+2f_2+4f_3+f_4)$$

6. 辛普生$\frac{3}{8}$公式。考虑闭区间$[x_0,x_3]$上的梯形公式：步长为 $3h$ 的 $T(f,3h)=(3h/2)(f_0+f_3)$，和步长为 h 的 $T(f,h)=(h/2)(f_0+2f_1+2f_2+f_3)$，证明：线性组合$(9T(f,h)-T(f,3h))/8$得到辛普生$\frac{3}{8}$公式。
7. 利用式(25)和式(26)建立式(27)。
8. 利用式(28)和式(29)建立式(30)。
9. 求下面的最小整数 K：

(a) $\int_0^2 8x^7\,dx = 256 \equiv R(K,K)$

(b) $\int_0^2 11x^{10}\,dx = 2048 \equiv R(K,K)$

10. 利用龙贝格积分，计算积分(i) $\int_0^1 \sqrt{x}dx$ 和(ii) $\int_0^1 2t^2\,dt$ 的逼近结果已在下表中给出：

(i)的逼近	(ii)的逼近
$R(0,0)=0.5000000$	$R(0,0)=1.0000000$
$R(1,1)=0.6380712$	$R(1,1)=0.6666667$
$R(2,2)=0.6577566$	$R(2,2)=0.6666667$
$R(3,3)=0.6636076$	$R(3,3)=0.6666667$
$R(4,4)=0.6655929$	$R(4,4)=0.6666667$

要求：

(a) 利用变量替换 $x=t^2$ 和 $dx=2t$，证明：两个积分有同样的数值。

(b) 讨论为什么积分(i)的收敛速度较慢而积分(ii)的收敛速度较快。

11. 基于中点公式的龙贝格积分。就效率和收敛速度而言，组合中点公式比组合梯形公式好。利用中点公式如下的事实：$\int_a^b f(x)\,dx = M(f,h)+E_M(f,h)$，公式 $M(f,h)$和误差项 $E_M(f,h)$由 $M(f,h)=h\sum_{k=1}^{N} f\left(a+\left(k-\frac{1}{2}\right)h\right)$ 给出，其中 $h=\frac{b-a}{N}$

和：

$$E_M(f,h)=a_1h^2+a_2h^4+a_3h^6+\cdots$$

要求：

(a) 由：

$$M(0)=\frac{b-a}{2}f\left(\frac{a+b}{2}\right)$$

推导连续中点公式：

$$M(J)=M(f,h_J)=h_J\sum_{k=1}^{2^J} f\left(a+\left(k-\frac{1}{2}\right)h_J\right)$$

其中 $h_J=\frac{b-a}{2^J}$。

(b) 给出如何用连续中点公式替换龙贝格积分中的连续梯形公式。

7.3.3 算法与程序

1. 利用程序 7.4 求习题 1 中的积分,精确到小数点后第 11 位。
2. 利用程序 7.4 求下面两个定积分,精确到小数点后第 10 位。两个定积分的精确值都是 π。解释两个龙贝格序列中积分速度的差别:

   (a) $\int_0^2 \sqrt{4x-x^2}\,dx$　　　(b) $\int_0^1 \frac{4}{1+x^2}dx$

3. 正态概率密度函数为 $f(t)=(1/\sqrt{2\pi})e^{-t^2/2}$,而累积分布(cumulative distribution)为由积分 $\Phi(x)=\frac{1}{2}+\frac{1}{\sqrt{2\pi}}\int_0^x e^{-t^2/2}dt$ 定义的函数。计算有 8 位有效数字的 $\Phi(0.5)$,$\Phi(1.0)$,$\Phi(1.5)$,$\Phi(2.0)$,$\Phi(2.5)$,$\Phi(3.0)$,$\Phi(3.5)$和 $\Phi(4.0)$的值。
4. 修改程序 7.3,使它在连续梯形公式的相邻值 $T(K-1)$和 $T(K)$相差小于 5×10^{-6}时中止。
5. 修改程序 7.3,使它能计算连续辛普生公式和布尔公式。
6. 修改程序 7.4,使它用连续中点公式进行龙贝格积分(利用习题 11 中的结果),利用程序求下面积分的近似值,精确到小数点后第 10 位。

   (a) $\int_0^1 \frac{\sin(x)}{x}dx$　　　(b) $\int_{-1}^1 \sqrt{1-x^2}\,dx$

7. 在程序 7.4 中,对给定定积分的逼近保存在一个下三角矩阵的对角线上,修改程序 7.4,使其能顺序计算龙贝格积分表的行,且结果保存在一个 $n\times1$ 的矩阵 R 中,从而节省空间。使用习题 1 来检验程序。

7.4 自适应积分

组合积分公式要求等距节点。典型情况下,在整个积分区间使用小步长 h,以保证整体精度。这并没有考虑到可能存在曲线的某些部分比其他部分变化剧烈的情况。引入一种方法,能在函数值变化大的部分减小步长,这是很有用的。该技术称为自适应积分,它的基础是辛普生公式。

辛普生公式使用$[a_k,b_k]$上的两个子区间:

$$S(a_k,b_k)=\frac{h}{3}(f(a_k)+4f(c_k)+f(b_k))\tag{1}$$

其中 $c_k=\frac{1}{2}[a_k+b_k]$,且 $h=(b_k-a_k)/2$。更进一步,若 $f\in C^4(a_k,b_k)$,则存在一个值 $d_1\in[a_k,b_k]$,使得:

$$\int_{a_k}^{b_k} f(x)\,dx = S(a_k,b_k)-h^5\frac{f^{(4)}(d_1)}{90}\tag{2}$$

7.4.1 区间细分(refinement)

区间$[a_k,b_k]$的4个子区间的组合辛普生公式可通过将该区间划分为两个相等子区间$[a_{k1},b_{k1}]$和$[a_{k2},b_{k2}]$,并在每段上递归地利用式(1)实现。只需要增加两个$f(x)$求值计算,其结果为:

$$S(a_{k1},b_{k1})+S(a_{k2},b_{k2})=\frac{h}{6}(f(a_{k1})+4f(c_{k1})+f(b_{k1}))+\frac{h}{6}(f(a_{k2})+4f(c_{k2})+f(b_{k2})) \tag{3}$$

其中,$a_{k1}=a_k$,$b_{k1}=a_{k2}=c_k$,$b_{k2}=b_k$,c_{k1}是$[a_{k1},b_{k1}]$的中点,而c_{k2}是$[a_{k2},b_{k2}]$的中点。在式(3)中,步长为$h/2$,它对应于等式右端的$h/6$。进一步,若$f\in C^4[a,b]$,则存在一个值$d_2\in[a_k,b_k]$,使得:

$$\int_{a_k}^{b_k}f(x)dx=S(a_{k1},b_{k1})+S(a_{k2},b_{k2})-\frac{h^5}{16}\frac{f^{(4)}(d_2)}{90} \tag{4}$$

设$f^{(4)}(d_1)\approx f^{(4)}(d_2)$,则可由式(2)和式(4)的右端得到关系:

$$S(a_k,b_k)-h^5\frac{f^{(4)}(d_2)}{90}\approx S(a_{k1},b_{k1})+S(a_{k2},b_{k2})-\frac{h^5}{16}\frac{f^{(4)}(d_2)}{90} \tag{5}$$

它可写为:

$$-h^5\frac{f^{(4)}(d_2)}{90}\approx\frac{16}{15}(S(a_{k1},b_{k1})+S(a_{k2},b_{k2})-S(a_k,b_k)) \tag{6}$$

将式(6)代入式(4),得误差估计:

$$\left|\int_{a_k}^{b_k}f(x)dx-S(a_{k1},b_{k1})-S(a_{k2},b_{k2})\right|\approx\frac{1}{15}\mid S(a_{k1},b_{k1})+S(a_{k2},b_{k2})-S(a_k,b_k)\mid \tag{7}$$

由于假设$f^{(4)}(d_1)\approx f^{(4)}(d_2)$,在使用该方法时,(7)式右端的$\frac{1}{15}$用$\frac{1}{10}$替换。这讲明下面的测试是合理的。

7.4.2 精度测试

设对区间$[a_k,b_k]$指定容差$\varepsilon_k>0$,若:

$$\frac{1}{10}\mid S(a_{k1},b_{k1})+S(a_{k2},b_{k2})-S(a_k,b_k)\mid<\varepsilon_k \tag{8}$$

我们推断,有:

$$\left|\int_{a_k}^{b_k}f(x)dx-S(a_{k1},b_{k1})-S(a_{k2},b_{k2})\right|<\varepsilon_k \tag{9}$$

于是利用辛普生组合公式(3)逼近积分:

$$\int_{a_k}^{b_k}f(x)dx\approx S(a_{k1},b_{k1})+S(a_{k2},b_{k2}) \tag{10}$$

且该逼近在区间$[a_k,b_k]$上的误差限为ε_k。

自适应积分通过应用辛普生公式(1)和(3)实现。从$\{[a_0,b_0],\varepsilon_0\}$开始,其中$\varepsilon_0$为$[a_0,b_0]$上数值积分的容差。该区间细分为两个子区间,记为$[a_{01},b_{01}]$和$[a_{02},b_{02}]$。若通过了精度测试式(8),则将积分公式(3)应用于区间$[a_0,b_0]$,过程结束;若未通过测试,则两个子区间记为$[a_1,b_1]$和$[a_2,b_2]$,在其上分别采用容差$\varepsilon_1=\frac{1}{2}\varepsilon_0$和$\varepsilon_2=\frac{1}{2}\varepsilon_0$。这样就得到两个子区间及其相应的容差,需要进一步细分和测试:$\{[a_1,b_1],\varepsilon_1\}$和$\{[a_2,b_2],\varepsilon_2\}$,其中$\varepsilon_1+\varepsilon_2=\varepsilon_0$。若必须继续进行自适应积分,则必须进一步细分子区间和进行测试,每个子区间都有相应的容差。

在第2步中我们首先考虑$\{[a_1,b_1],\varepsilon_1\}$,并将区间$[a_1,b_1]$细分为$[a_{11},b_{11}]$和$[a_{12},b_{12}]$,若它们以容差$\varepsilon_1$通过了精度测试式(8),则在区间$[a_1,b_1]$上应用公式(3),且在此区间上能保证精度;若不能以容差ε_1通过精度测试式(8),则必须对两个子区间$[a_{11},b_{11}]$和$[a_{12},b_{12}]$进行细分,并在第3步中以减小的容差$\frac{1}{2}\varepsilon_1$进行测试。此外,第2步还要考虑$\{[a_2,b_2],\varepsilon_2\}$,将区间$[a_2,b_2]$细分为子区间$[a_{21},b_{21}]$和$[a_{22},b_{22}]$。若以容差$\varepsilon_2$通过精度测试式(8),则在区间$[a_2,b_2]$上应用公式(3),且在此区间上能保证精度;如果以容差ε_2测试式(8),则必须对两个子区间$[a_{21},b_{21}]$和$[a_{22},b_{22}]$再进行细分,并在第3步中以减小的容差$\frac{1}{2}\varepsilon_2$进行测试。因此,第2步生成3或4个子区间,我们对其进行连续地标注。3个子区间重新标记为$\{[a_1,b_1],\varepsilon_1\}$,$\{[a_2,b_2],\varepsilon_2\}$,$\{[a_3,b_3],\varepsilon_3\}$。其中:$\varepsilon_1+\varepsilon_2+\varepsilon_3=\varepsilon_0$。对于4个子区间,则为$\{[a_1,b_1],\varepsilon_1\}$,$\{[a_2,b_2],\varepsilon_2\}$,$\{[a_3,b_3],\varepsilon_3\}$和$\{[a_4,b_4],\varepsilon_4\}$。其中:$\varepsilon_1+\varepsilon_2+\varepsilon_3+\varepsilon_4=\varepsilon_0$。

若必须继续进行自适应积分,必须以各自相应的容差测试较小的区间。式(4)中的误差项显示,每一次对一个较小的区间进行细分,误差的衰减因子大约为$\frac{1}{16}$。这样该过程将在有限步之后停止。该方法的记录中包括一个标记变量,用以指示每个子区间是否通过了精度测试。为避免不必要的$f(x)$求值计算,函数值可以在对应于每个子区间的一个数据表中给出。该过程的细节将在程序7.6中给出。

例7.16 用自适应积分求定积分$\int_0^4 13(x-x^2)e^{-3x/2}dx$的数值逼近,起始容差为$\varepsilon_0=0.00001$。

该方法的实现需要20个子区间,表7.8列出了每个子区间$[a_k,b_k]$,组合辛普生公式$S(a_{k1},b_{k1})+S(a_{k2},b_{k2})$,该逼近的误差界以及相应的容差$\varepsilon_k$。通过对辛普生公式逼近求和得到积分的近似值:

$$\int_0^4 13(x-x^2)e^{-3x/2}dx \approx -1.54878823413 \tag{11}$$

表7.8 $f(x)=13(x-x^2)e^{-3x/2}$的自适应积分计算

a_k	b_k	$S(a_{k1},b_{k1})+S(a_{k2},b_{k2})$	(8)的左误差界	$[a_k,b_k]$的容差ε_k
0.0	0.0625	0.02287184840	0.00000001522	0.00000015625
0.0625	0.125	0.05948686456	0.00000001316	0.00000015625
0.125	0.1875	0.08434213630	0.00000001137	0.00000015625

（续表）

a_k	b_k	$S(a_{k1},b_{k1})+S(a_{k2},b_{k2})$	(8)的左误差界	$[a_k,b_k]$的容差 ϵ_k
0.1875	0.25	0.09969871532	0.00000000981	0.00000015625
0.25	0.375	0.21672136781	0.00000025055	0.0000003125
0.375	0.5	0.20646391592	0.00000018402	0.0000003125
0.5	0.625	0.17150617231	0.00000013381	0.0000003125
0.625	0.75	0.12433363793	0.00000009611	0.0000003125
0.75	0.875	0.07324515141	0.00000006799	0.0000003125
0.875	1.0	0.02352883215	0.00000004718	0.0000003125
1.0	1.125	−0.02166038952	0.00000003192	0.0000003125
1.125	1.25	−0.06065079384	0.00000002084	0.0000003125
1.25	1.5	−0.21080823822	0.00000031714	0.000000625
1.5	2.0	−0.60550965007	0.00000003195	0.00000125
2.0	2.25	−0.31985720175	0.00000008106	0.000000625
2.25	2.5	−0.30061749228	0.00000008301	0.000000625
2.5	2.75	−0.27009962412	0.00000007071	0.000000625
2.75	3.0	−0.23474721177	0.00000005447	0.000000625
3.0	3.5	−0.36389799695	0.00000103699	0.00000125
3.5	4.0	−0.24313827772	0.00000041708	0.00000125
总计		−1.54878823413	0.00000296809	0.00001

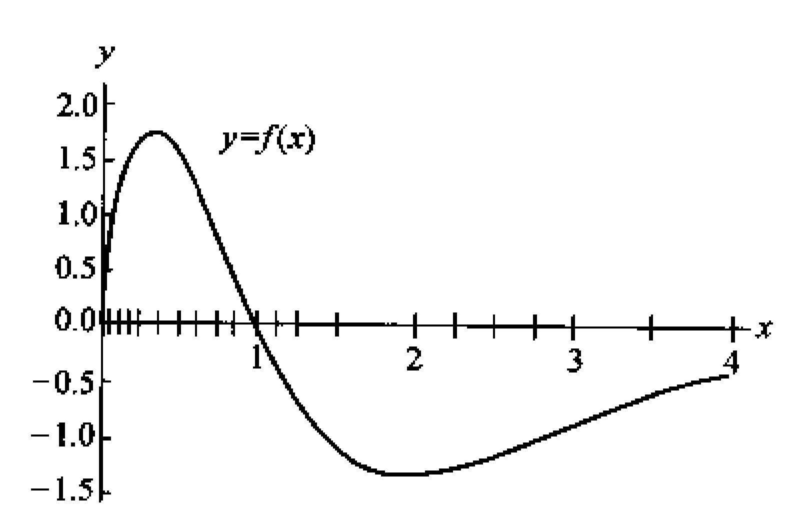


图 7.9　自适应积分中[0,4]的子区间

积分的真值为：

$$\int_0^4 13(x-x^2)e^{-3x/2}dx = \frac{4108e^{-6}-52}{27} \tag{12}$$

$$= -1.5487883725279481333$$

故，自适应积分的误差为：

$$|-1.54878837253-(-1.54878823413)| = 0.00000013840 \tag{13}$$

小于给定的容差 $\epsilon_0=0.00001$。自适应方法包含了区间[0,4]的 20 个子区间，用了 81 次函数求值。图 7.9 显示了 $y=f(x)$ 曲线和 20 个子区间，在原点附近函数值变化大的部分区间宽度较小。在自适应方法的区间细分和精度测试过程中，前 4 个宽度为 0.25 的区间被二分为宽度为 0.03125 的 8 个子区间。若继续使用该步长，则需要 $M=128$ 个子区间来进行组合辛普生公式的计算，其近似结果为 −1.54878844029，误差值为 0.0000006776。虽然组合辛普生公式的误差将近是自适应积分方法误差的一半，但它增加了 176 个函数求

值计算,而精度的提高微不足道,故自适应积分的计算量节省是显著的。

程序 7.5 中 srule 是对 7.1 节中辛普生公式的改进,其输出为包含区间[$a0$, $b0$]上辛普生公式结果的向量 Z。程序 7.6 调用 srule 作为一个子程序来实现自适应积分过程中产生的子区间上的辛普生公式。

程序 7.5 (辛普生公式) 用辛普生公式逼近积分:

$$\int_{a0}^{b0} f(x)\,dx \approx \frac{h}{3}f(a_0) + 4f(c0) + f(b0))$$

其中 $c0 = (a0 + b0)/2$

```
function Z = srule(f,a0,b0,tol0)
% Input   - f is the integrand input as a string 'f'
%         - a0 and b0 are upper and lower limits of integration
%         - tol0 is the tolerance
% Output  - Z is a 1x6 vector [a0 b0 S S2 err tol1]
h = (b0 - a0)/2;
C = zeros(1,3);
C = feval(f,[a0(a0 + b0)/2 b0]);
S = h * (C(1) + 4 * C(2) + C(3))/3;
S2 = S;
tol1 = tol0;
err = tol0;
Z = [a0 b0 S S2 err tol1];
```

程序 7.6 (用辛普生公式的自适应积分) 逼近积分:

$$\int_a^b f(x)\,dx \approx \sum_{k=1}^{M}(f(x_{4k-4}) + 4f(x_{4k-3}) + 2f(x_{4k-2}) + 4f(x_{4k-1}) + f(x_{4k})$$

在 $4M$ 个子区间$[x_{4k-4}, x_{4k}]$上应用组合辛普生公式,其中$[a, b] = [x_0, x_{4M}]$而 $x_{4k-4+j} = x_{4k-4} + jh_k$, $k = 1,\cdots,M$ 和$j = 1,\cdots,4$

```
function [SRmat,quad,err] = adapt(f,a,b,tol)
% Input   - f is the integrand input as a string 'f'
%         - a and b are upper and lower limits of integration
%         - tol is the tolerance
% Output  - SRmat is the table of values
%         - quad is the quadrature value
%         - err is the error estimate
% Initialize values
SRmat = zeros(30,6)
iterating = 0;
done = 1;
SRvec = zeros(1,6);
SRvec = srule(f,a,b,tol);
SRmat = (1,1:6) = SRvec;
m = 1;
```

```
state = iterating;
while(state == iterating)
    n = m;
    for j = n: -1:1
      p = j;
      SROvec = SRmat(p,:);
      err = SR0vec(5);
      tol = SR0vec(6);
      if (tol <= err)
         % Bisect interval,apply Simpson's rule
         % recursively,and determine error
         state = done;
         SR1vec = SR0vec;
         SR2vec = SR0vec;
         a = SR0vec(1);
         b = SR0vec(2);
         c = (a + b)/2;
         err = SR0vec(5);
         tol = SR0vec(6);
         tol2 = tol/2;
         SR1vec = srule(f,a,c,tol2);
         SR2vec = srule(f,c,b,tol2);
         err = abs(SR0vec(3) - SR1vec(3) - SR2vec(3))/10;
         % Accuracy test
         if (err < tol)
            SRmat(p,:) = SR0vec;
            SRmat(p,4) = SR1vec(3) + SR2vec(3);
            SRmat(p,5) = err;
        else
            SRmat(p + 1:m + 1,:) = SRmat(p:m,:);
            m = m + 1;
            SRmat(p,:) = SR1vec;
            SRmat(p + 1,:) = SR2vec;
            state = iterating;
        end
            end
        end
    end
    quad = sum(SRmat(:,4));
    err = sum(abs(SRmat(:,5)));
    SRmat = SRmat(1:m,1:6);
```

7.4.3 算法与程序

1. 用程序 7.6 求以下定积分的近似值，使用起始容差 $\varepsilon_0 = 0.00001$。

(a) $\int_0^3 \frac{\sin(2x)}{1+x^5}dx$ (b) $\int_0^3 \sin(4x)e^{-2x}dx$

(c) $\int_{0.04}^1 \frac{1}{\sqrt{x}}dx$ (d) $\int_0^2 \frac{1}{x^2+\frac{1}{10}}dx$

(e) $\int_{1/(2\pi)}^2 \sin\left(\frac{1}{x}\right)dx$ (f) $\int_0^2 \sqrt{4x-x^2}\,dx$

2. 对问题1中的每个定积分，绘制一个类似于图7.9的图。提示：SRmat的第一列包含了自适应积分过程的子区间端点(除了 b 以外)。如果 T = SRmat(:,1) 和 Z = zeros(length(T))'，则 plot(T,Z,'.') 将生成子区间(除了右端点 b 以外)。
3. 修改程序7.6以在每个子区间 $[a_k,b_k]$ 上应用布尔公式。
4. 使用问题3中修改后的程序，计算问题1中定积分的近似值，并绘制类似于图7.9的图。

7.5 高斯－勒让德积分(可选)

我们希望计算曲线：

$$y=f(x),\ -1\leqslant x\leqslant 1$$

下的面积。若只允许进行两次函数求值，什么方法能产生最好的答案呢？我们已经看到，梯形公式是计算曲线下面积且在端点 $(-1,f(-1))$ 和 $(1,f(1))$ 处求两次函数的方法。但是若 $y=f(x)$ 的曲线为向下凹的，则逼近的误差为曲线和连接两个端点直线之间区域的面积，另一个例子在图7.10(a)中给出。

如果能用区间 $[-1,1]$ 中的节点 x_1 和 x_2，过点 $(x_1,f(x_1))$ 和 $(x_2,f(x_2))$ 的直线，则直线下的面积更接近曲线下的面积(见图7.10(b))。直线方程为：

$$y=f(x_1)+\frac{(x-x_1)(f(x_2)-f(x_1))}{x_2-x_1} \tag{1}$$

而直线下的梯形面积为：

$$A_{trap}=\frac{2x_2}{x_2-x_1}f(x_1)-\frac{2x_1}{x_2-x_1}f(x_2) \tag{2}$$

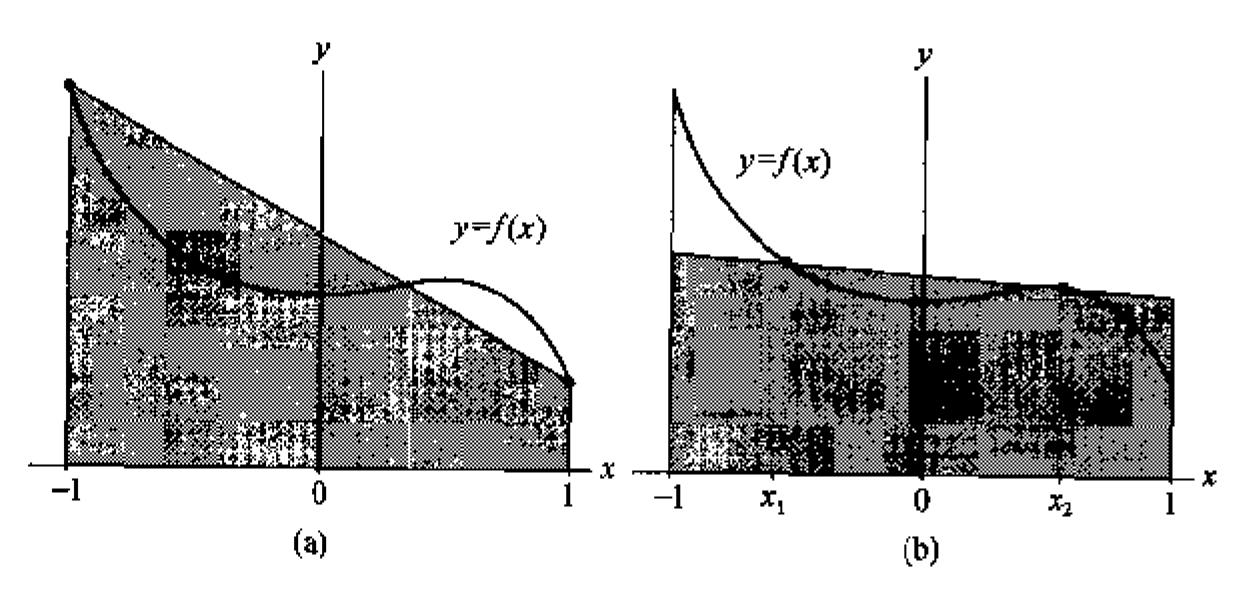


图7.10 (a)用横坐标 -1 和 1 的梯形逼近
(b)用横坐标 x_1 和 x_2 的梯形逼近

注意：梯形公式是式(2)的一种特例，当我们选择 $x_1=-1, x_2=1$ 和 $h=2$ 时，有：

$$T(f,h)=\frac{2}{2}f(x_1)-\frac{-2}{2}f(x_2)=f(x_1)+f(x_2)$$

用待定系数法找出横坐标 x_1,x_2，权 ω_1 和 ω_2，使得公式：

$$\int_{-1}^{1}f(x)dx\approx\omega_1 f(x_1)+\omega_2 f(x_2) \tag{3}$$

对 3 次多项式(即 $f(x)=a_3x^3+a_2x^2+a_1x+a_0$)精确。由于式(3)中有 4 个系数 ω_1,ω_2,x_1 和 x_2 待定,我们可选择 4 个条件来满足。利用积分的可加性,只需使式(3)对 4 个函数 $f(x)=1$, x^2,x^3 精确即可。4 个积分条件为:

$$\begin{aligned} f(x)&=1: && \int_{-1}^{1} 1\,dx = 2 = \omega_1+\omega_2 \\ f(x)&=x: && \int_{-1}^{1} x\,dx = 0 = \omega_1x_1+\omega_2x_2 \\ f(x)&=x^2: && \int_{-1}^{1} x^2\,dx = \frac{2}{3} = \omega_1x_1^2+\omega_2x_2^2 \\ f(x)&=x^3: && \int_{-1}^{1} x^3\,dx = 0 = \omega_1x_1^3+\omega_2x_2^3 \end{aligned} \tag{4}$$

求解非线性方程组:

$$\omega_1+\omega_2=2 \tag{5}$$

$$\omega_1x_1=-\omega_2x_2 \tag{6}$$

$$\omega_1x_1^2+\omega_2x_2^2=\frac{2}{3} \tag{7}$$

$$\omega_1x_1^3=-\omega_2x_2^3 \tag{8}$$

用式(6)除式(8),得:

$$x_1^2=x_2^2 \text{ 或 } x_1=-x_2 \tag{9}$$

由式(9),并将(6)式左端被 x_1 除,右端被 $-x_2$ 除,得:

$$\omega_1=\omega_2 \tag{10}$$

将式(10)代人式(5),结果为 $\omega_1+\omega_2=2$,故:

$$\omega_1=\omega_2=1 \tag{11}$$

在式(7)中用式(9)和式(11),可写出:

$$\omega_1x_1^2+\omega_2x_2^2=x_2^2+x_2^2=\frac{2}{3} \quad \text{或} \quad x_2^2=\frac{1}{3} \tag{5}$$

最后,由式(12)和式(9)可知节点为:

$$-x_1=x_2=1/3^{1/2}\approx 0.5773502692$$

我们已经找到了两点高斯-勒让德公式的节点和权。由于该公式对 3 次多项式精确,因此误差项包含 4 阶导数。对误差项的讨论可在参考文献[41]中找到。

定理 7.8 (两点高斯-勒让德公式) 若 f 在 $[-1,1]$ 内连续,则:

$$\int_{-1}^{1} f(x)\,dx \approx G_2(f) = f\left(\frac{-1}{\sqrt{3}}\right)+f\left(\frac{1}{\sqrt{3}}\right) \tag{13}$$

高斯-勒让德公式 $G_2(f)$ 的精度为 $n=3$。若 $f\in C^4[-1,1]$,则:

$$\int_{-1}^{1} f(x)\,dx = f\left(\frac{-1}{\sqrt{3}}\right)+f\left(\frac{1}{\sqrt{3}}\right)+E_2(f) \tag{14}$$

其中:

$$E_2(f)=\frac{f^{(4)}(c)}{135} \tag{15}$$

例 7.17 利用两点高斯－勒让德公式计算逼近：

$$\int_{-1}^{1}\frac{dx}{x+2}=\ln(3)-\ln(1)\approx 1.09861$$

并将结果与 $h=2$ 的梯形公式 $T(f,h)$和 $h=1$ 的辛普生公式比较。

设 $S(f,h)$表示两点高斯－勒让德公式，则 $G_2(f)$：

$$G_2(f)=f(-0.57735)+f(0.57735)$$
$$=0.70291+0.38800=1.09091$$

$$T(f,2)=f(-1.00000)+f(1.00000)$$
$$=1.00000+0.33333=1.33333$$

$$S(f,1)=\frac{f(-1)+4f(0)+f(1)}{3}=\frac{1+2+\frac{1}{3}}{3}=1.11111$$

误差分别为 0.00770，－0.23472 和 －0.01150，由此可见，高斯－勒让德公式效果最佳。注意高斯－勒让德公式只需要 2 次函数求值，而辛普生公式需要 3 次。在本例中 $G_2(f)$的误差规模约为 $S(f,1)$误差规模的 61%。

一般的 N－点高斯－勒让德公式对次数小于等于 $2N-1$ 的多项式是精确的，数值积分公式为：

$$G_N(f)=\omega_{N,1}f(x_{N,1})+\omega_{N,2}f(x_{N,2})+\cdots+\omega_{N,N}f(x_{N,N}) \tag{16}$$

所需要的横坐标 $x_{N,k}$和权$\omega_{N,k}$已制成表，便于查找；表 7.9 列出了直至 8 点的值，该表中还包含了对应于 $G_N(f)$的误差项 $E_N(f)$的形式，可用来确定高斯－勒让德积分公式的精度。

表 7.9 高斯－勒让德节点和权

$$\int_{-1}^{1}f(x)\,dx=\sum_{k=1}^{N}\omega_{N,k}f(x_{N,k})+E_N(f)$$

N	横坐标 $x_{N,k}$	权 $\omega_{N,k}$	截断误差 $E_N(f)$
2	−0.5773502692 0.5773502692	1.0000000000 1.0000000000	$\frac{f^{(4)}(c)}{135}$
3	±0.7745966692 0.0000000000	0.5555555556 0.8888888888	$\frac{f^{(6)}(c)}{15\,750}$
4	±0.8611363116 ±0.3399810436	0.3478548451 0.6521451549	$\frac{f^{(8)}(c)}{3\,472\,875}$
5	±0.9061798459 ±0.5384693101 0.0000000000	0.2369268851 0.4786286705 0.5688888888	$\frac{f^{(10)}(c)}{1\,237\,732\,650}$
6	±0.9324695142 ±0.6612093865 ±0.2386191861	0.1713244924 0.3607615730 0.4679139346	$\frac{f^{(12)}(c)2^{13}(6!)^4}{(12!)^3 13!}$
7	±0.9491079123 ±0.7415311856 ±0.4058451514 0.0000000000	0.1294849662 0.2797053915 0.3818300505 0.4179591837	$\frac{f^{(14)}(c)2^{15}(7!)^4}{(14!)^3 15!}$
8	±0.9602898565 ±0.7966664774 ±0.5255324099 ±0.1834346425	0.1012285363 0.2223810345 0.3137066459 0.3626837834	$\frac{f^{(16)}(c)2^{17}(8!)^4}{(16!)^3 17!}$

表7.9中的值没有简洁的表示形式，这使该方法在手算时不甚有吸引力；但当这些值保存在计算机中时，在需要时查找甚为便捷。节点实际是勒让德多项式的根，而对应的权要通过求解方程组得到。三点高斯－勒让德公式的节点是 $-(0.6)^{1/2}$、0 和 $(0.6)^{1/2}$，对应的权为 5/9、8/9 和 5/9 。

定理 7.9（三点高斯－勒让德公式） 若 f 在 $[-1,1]$ 内连续，则：

$$\int_{-1}^{1} f(x)\,dx \approx G_3(f) = \frac{5f(-\sqrt{3/5}) - 8f(0) + 5f(\sqrt{3/5})}{9} \tag{17}$$

高斯－勒让德公式 $G_3(f)$ 的精度为 $n=5$。若 $f\in C^6[-1,1]$，则：

$$\int_{-1}^{1} f(x)\,dx = \frac{5f(-\sqrt{3/5}) + 8f(0) + 5f\sqrt{3/5}}{9} + E_3(f) \tag{18}$$

其中：

$$E_3(f) = \frac{f^{(6)}(c)}{15\ 750} \tag{19}$$

例 7.18 证明三点高斯－勒让德公式对：

$$\int_{-1}^{1} 5x^4\,dx = 2 = G_3(f)$$

是精确的。

由于被积函数为 $f(x)=5x^4$，而 $f^{(6)}(x)=0$，可由式(19)得 $E_3(f)=0$。但在本例中用式(17)进行计算更有启发性：

$$G_3(f) = \frac{5(5)(0.6)^2 + 0 + 5(5)(0.6)^2}{9} = \frac{18}{9} = 2$$

下面的结果说明如何改变积分变量，使得高斯－勒让德公式可在区间 $[a,b]$ 内应用。

定理 7.10 设区间 $[-1,1]$ 内的 N 点高斯－勒让德公式的横坐标 $\{x_{N,K}\}_{k=1}^{N}$ 和权 $\{\omega_{N,k}\}_{k=1}^{N}$ 已知，欲在区间 $[a,b]$ 内应用公式，使用变量替换：

$$t = \frac{a+b}{2} + \frac{b-a}{2}x \text{ 和 } dt = \frac{b-a}{2}dx \tag{20}$$

则通过如下关系：

$$\int_a^b f(t)\,dt = \int_{-1}^{1} f\left(\frac{a+b}{2} + \frac{b-a}{2}x\right)\frac{b-a}{2}\,dx \tag{21}$$

可得积分公式：

$$\int_a^b f(t)\,dt = \frac{b-a}{2}\sum_{k=1}^{N}\omega_{N,k} f\left(\frac{a+b}{2} + \frac{b-a}{2}x_{N,k}\right) \tag{22}$$

例 7.19 用三点高斯－勒让德公式逼近：

$$\int_1^5 \frac{dt}{t} = \ln(5) - \ln(1) \approx 1.609438$$

并将结果与 $h=1$ 的布尔公式 $B(2)$ 比较。

这里 $a=1$ 而 $b=5$，故由式(22)得：

$$G_3(f) = (2)\frac{5f(3-2(0.6)^{1/2}) + 8f(3+0) + 5f(3+2(0.6)^{1/2})}{9}$$

$$= (2)\frac{3.446359 + 2.666667 + 1.099096}{9} = 1.602694$$

在例 7.13 中我们知道布尔公式的结果为 $B(2) = 1.617778$,误差分别为 0.006744 和 -0.008340,故在此情况下高斯 - 勒让德公式略优,注意高斯 - 勒让德公式只需要求 3 次函数值,而布尔公式需 5 次。本例中两个误差规模相同。

高斯 - 勒让德积分公式非常精确,在需对许多性质相同的积分求值时,需要认真考虑。在这种情况下,应该如下进行:选出几个有代表性的积分,包括可能出现最坏情况的积分;确定获得需要精度所需的采样点数 N;然后固定 N,对所有积分用 N 个采样点由高斯 - 勒让德公式计算。

对给定值 N,程序 7.7 要求将表 7.9 中的横坐标和权值分别保存在 $1xN$ 的矩阵 A 和 W 中,这可在 MATLAB 的命令窗口中实现,或将矩阵存为 M 文件。可以将表 7.9 保存在 35×2 的矩阵 G 中,G 的第一行包含横坐标,第二行包含对应的权。则对给定的值 N,矩阵 A 和 W 为 G 的子矩阵。例如,若 $N = 3$,则 $A = G(3:5,1)'$ 且 $W = G(3:5,2)'$。

程序 7.7 (高斯 - 勒让德求积分公式) 利用 $f(x)$ 在 N 个非等步长点 $\{t_{N,k}\}_{k=1}^{N}$ 的采样求积分:

$$\int_a^b f(x)\,dx \approx \frac{b-a}{2}\sum_{k=1}^{N}\omega_{N,k}f(t_{N,k})$$

的逼近。使用变量替换:

$$t = \frac{a+b}{2} + \frac{b-a}{2}x \text{ 和 } dt = \frac{b-a}{2}dx$$

横坐标 $\{x_{N,k}\}_{k=1}^{N}$ 和权 $\{\omega_{N,k}\}_{k=1}^{N}$ 必须从一个表中获得

```
function quad = gauss(f,a,b,A,W)
% Input    - f is the integrand input as a string 'f'
%          - a and b are upper and lower limits of integration
%          - A is the 1 x N vector of abscissas from Table 7.9
%          - W is the 1 x N vector of weights from Table 7.9
% Output   - quad is the quadrature value
N = length(A);
T = zeros(1,N);
T = ((a + b)/2) + ((b - a)/2) * A
quad = ((b - a)/2) * sum(W. * feval(f,T));
```

7.5.1 习题

在习题 1 ~ 5 中,证明两个积分是等价的,并计算 $G_2(f)$。

1. $\int_0^2 6t^5\,dt = \int_{-1}^{1} 6(x+1)^5\,dx$

2. $\int_0^2 \sin(t)\,dt = \int_{-1}^{1} \sin(x+1)\,dx$

3. $\int_0^1 \frac{\sin(t)}{t}dt = \int_{-1}^1 \frac{\sin((x+1)/2)}{x+1}dx$

4. $\frac{1}{\sqrt{2\pi}}\int_0^1 e^{-t^2/2}dt = \frac{1}{\sqrt{2\pi}}\int_{-1}^1 \frac{e^{-(x+1)^2/8}}{2}dx$

5. $\frac{1}{\pi}\int_0^\pi \cos(0.6\sin(t))dt = 0.5\int_{-1}^1 \cos\left(0.6\sin\left((x+1)\frac{\pi}{2}\right)\right)dx$

6. 利用表 7.9 中的 $E_N(f)$和定理 7.10 中的变量代换求出最小整数 N,使得对:

(a) $\int_0^2 8x^7 dx = 256 = G_N(f)$

(b) $\int_0^2 11x^{10} dx = 2048 = G_N(f)$

有 $E_N(f)=0$。

7. 求下列勒让德多项式的根,并将它们与表 7.9 中的横坐标相比较。

(a) $P_2(x)=(3x^2-1)/2$

(b) $P_3(x)=(5x^3-3x)/2$

(c) $P_4(x)=(35x^4-30x^2+3)/8$

8. 闭区间$[-1,1]$内两点高斯-勒让德公式的截断误差项为 $f^{(4)}(c_1)/135$,$[a,b]$内辛普生公式的截断误差为 $-h^5f^{(4)}(c_2)/90$,试比较当$[a,b]=[-1,1]$时的两个截断误差,你认为哪个更好,为什么?

9. 三点高斯-勒让德公式为:

$$\int_{-1}^1 f(x)dx \approx \frac{5f(-(0.6)^{1/2})+8f(0)+5f((0.6)^{1/2})}{9}$$

证明:公式对 $f(x)=1,x,x^2,x^3,x^4,x^5$ 是精确的。提示:若 f 为奇函数 (即,$f(-x)=f(x)$),则 f 在$[-1,1]$内的积分为 0。

10. 三点高斯-勒让德公式在区间$[-1,1]$内的截断误差为 $f^{(6)}(c_1)/15\,750$,$[a,b]$内布尔公式的截断误差为 $-8h^7f^{(6)}(c^2)/945$,比较当$[a,b]=[-1,1]$时的误差项,哪种方法更好?为什么?

11. 用以下步骤推导三点高斯-勒让德公式。利用横坐标是三次勒让德多项式的根:

$$x_1=-(0.6)^{1/2}, x_2=0, x_3=(0.6)^{1/2}。$$

求权 ω_1,ω_2 和 ω_3,使得:

$$\int_{-1}^1 f(x)dx \approx \omega_1 f(-(0.6)1/2)+\omega_2 f(0)+\omega_3 f((0.6)^{1/2})$$

对函数 $f(x)=1,x$ 和 x^2 精确。提示:首先得出线性方程组:

$$\omega_1+\omega_2+\omega_3=2$$

$$-(0.6)^{1/2}\omega_1+(0.6)^{1/2}\omega_3=0$$

$$0.6\omega_1+0.6\omega_3=\frac{2}{3}$$

然后求解。

12. 在实际运算中,若要对许多同一类型的积分求值,需要先进行初步分析,来确定获得需

要的精度所需的函数求值数。假设需要 17 次求值,比较龙贝格积分结果 $R(4,4)$ 和高斯－勒让德结果 $G_{17}(f)$。

7.5.2 算法与程序

1. 对习题 1～5 中的每个积分,用程序 7.7 求 $G_6(f)$, $G_7(f)$ 和 $G_8(f)$。
2. (a) 修改程序 7.7,使之可计算 $G_1(f)$, $G_2(f)$, $\cdots$, $G_8(f)$,并当结果逼近 $G_{N-1}(f)$ 和 $G_N(f)$ 的相对误差小于预设值 tol,即:

$$\frac{2|G_{N-1}(f)-G_N(f)|}{|G_{N-1}(f)+G_N(f)|}<\text{tol}$$

时停止。

提示:如同上节最后所讨论的,将表 7.9 以 35×2 矩阵 G 保存在一个 M 文件中。

(b) 利用 (a) 中的程序逼近习题 1～5 中的积分,精确到小数点后第 5 位。

3. (a) 使用 6 点高斯－勒让德公式逼近积分:

$$v(x) = x^2 + 0.1\int_0^3 (x^2+t)v(t)\,dt$$

将逼近解代入积分式的右边并简化之。

(b) 使用 8 点高斯－勒让德公式重做(a)。

第8章 数值优化

在机械工程的长方形三维空间模拟振动过程常使用二维波动方程。如果将平板的四个边固定,则正弦振荡可用双重傅里叶级数表示。设在某一个特定的时刻,在点(x,y)处的高度$z=f(x,y)$可用下式表示:

$$z = f(x,y) = 0.02\sin(x)\sin(y) - 0.03\sin(2x)\sin(y) + 0.04\sin(x)\sin(2y) + 0.08\sin(2x)\sin(2y)$$

最大偏转点位于何处?分别观察图8.1(a)和(b)中的三维图形和等高线图,可看出在区间$0\leqslant x\leqslant\pi, 0\leqslant y\leqslant\pi$内有两个局部极小值和两个局部极大值。可以通过数值方法找到它们的大致位置:

$$f(0.8278,2.3322) = -0.1200 \text{ 和 } f(2.5351,0.6298) = -0.0264$$

是局部极小值,而:

$$f(0.9241,0.7640) = 0.0998 \text{ 和 } f(2.3979,2.2287) = 0.0853$$

是局部极大值。

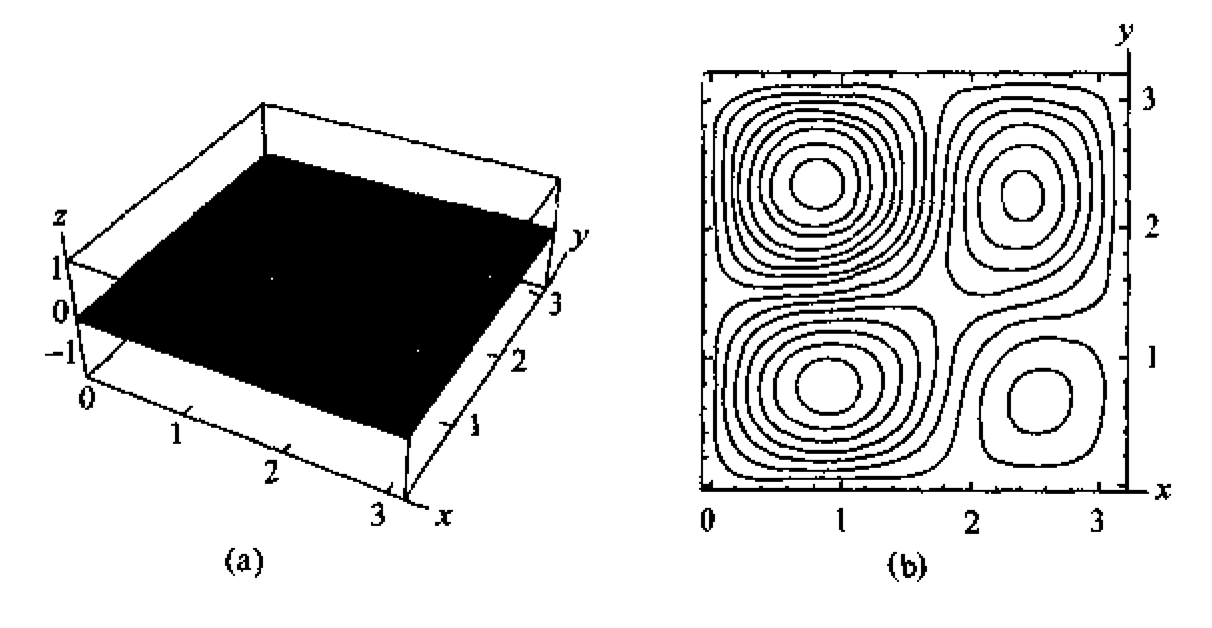


图8.1 (a) 振荡平板的位移$z=f(x,y)$;(b) 振荡平板的等高线$f(x,y)=C$

本章将简单介绍求解包含一个或多个变量的函数极值的基本方法。

8.1 函数极小值

定义8.1(局部极值) 设有一个函数f,如果存在一个包含p的开区间I,对所有$x\in I$满足$f(p)\leqslant f(x)$,则称函数f在$x=p$处有一个局部极小值。同理,如果对所有$x\in I$满足$f(x)\leqslant f(p)$,则称函数f在$x=p$处有一个局部极大值。如果f在$x=p$处有局部极小值或局部极大值,则称f在$x=p$处有一个局部极值。

定义8.2(递增和递减) 设函数$f(x)$在区间I内有定义。

(i) 对所有$x_1,x_2\in I$,如果$x_1<x_2$,则$f(x_1)<f(x_2)$,则称函数f在区间I内是递增的。

(ii) 对所有$x_1,x_2\in I$,如果$x_1<x_2$,则$f(x_1)>f(x_2)$,则称函数f在区间I内是递减的。

定理 8.1 设 $f(x)$ 在区间 $I=[a,b]$ 内连续,且在区间 (a,b) 内可微。

(i) 对所有 $x\in(a,b)$,如果 $f'(x)>0$,则 $f(x)$ 在区间 I 内递增。

(ii)对所有 $x\in(a,b)$,如果 $f'(x)<0$,则 $f(x)$ 在区间 I 内递减。

定理 8.2 设 $f(x)$ 在区间 $I=[a,b]$ 内有定义,且在内部点 $p\in(a,b)$ 处有一个局部极值。如果 $f(x)$ 在 $x=p$ 处可微,则 $f'(p)=0$。

定理 8.3(一阶导数测试) 设 $f'(x)$ 在区间 $I=[a,b]$ 内连续。而且除了在 $x=p$ 处,对所有 $x\in(a,b)$ 有定义。

(i) 如果在区间 (a,p) 内有 $f'(x)<0$ 且在区间 (p,b) 内有 $f'(x)>0$,则 $f(p)$ 有一个局部极小值。

(ii) 如果在区间 (a,p) 内有 $f'(x)>0$ 且在区间 (p,b) 内有 $f'(x)<0$,则 $f(p)$ 有一个局部极大值。

定理 8.4(二阶导数测试) 设 $f(x)$ 在区间 $[a,b]$ 内连续,且 f' 和 f'' 在区间 (a,b) 内有定义,且存在 $p\in(a,b)$ 是满足 $f'(p)=0$ 的一个临界点。

(i) 如果 $f''(p)>0$,则 $f(p)$ 是函数 f 的局部极小值。

(ii) 如果 $f''(p)<0$,则 $f(p)$ 是函数 f 的局部极大值。

(iii) 如果 $f''(p)=0$,则测试不确定。

例 8.1 根据二阶导数测试,对在区间 $[-2,2]$ 内的 $f(x)=x^3+x^2-x+1$ 的局部极值进行分类。

解:

函数的一阶导数为 $f'(x)=3x^2+2x-1=(3x-1)(x+1)$,二阶导数为 $f''(x)=6x+2$。有两个点满足 $f'(x)=0$(即 $x=1/3,-1$)。

情况 (i): 在 $x=1/3$ 处,可得到 $f'(1/3)=0$ 和 $f''(1/3)=4>0$,因此,$f(x)$ 在 $x=1/3$ 处有一局部极小值。

情况(ii): 在 $x=-1$ 处,可得到 $f'(-1)=0$ 和 $f''(-1)=-4<0$,因此,$f(x)$ 在 $x=-1$ 处有一局部极大值。

8.1.1 搜索方法

另一个求解 $f(x)$ 极值的方法是对函数进行多次估计来搜索一局部极小值。为了减少函数估计次数,选择在何处对 $f(x)$ 进行估计是非常重要的。其中一个最有效的方法是黄金分割搜索法,它是根据搜索区间的比例命名的。

1. 黄金分割率

设初始区间为 $[0,1]$。如果 $0.5<r<1$,则 $0<1-r<0.5$,且将区间分成三部分 $[0,1-r]$,$[1-r,r]$ 和 $[r,1]$。一个判定过程决定是从右边压缩得到新的区间 $[0,r]$,或是从左边压缩得到新的区间 $[1-r,1]$。然后将新的子区间按前面同样的比例再分成三部分。

选择 r 使得其中一个旧的点位于新区间中的正确位置,如图 8.2 所示。这表示比例 $(1-r):r$ 与 $r:1$ 相同。因此 r 满足等式 $1-r=r^2$,它也可表示为一个二次方程 $r^2+r-1=0$。

满足 $0.5 < r < 1$ 的解为 $r = (\sqrt{5} - 1)/2$。

为了用黄金分割搜索法求 $f(x)$的最小值,必须满足一个特殊的条件,以保证在区间内确有最小值。

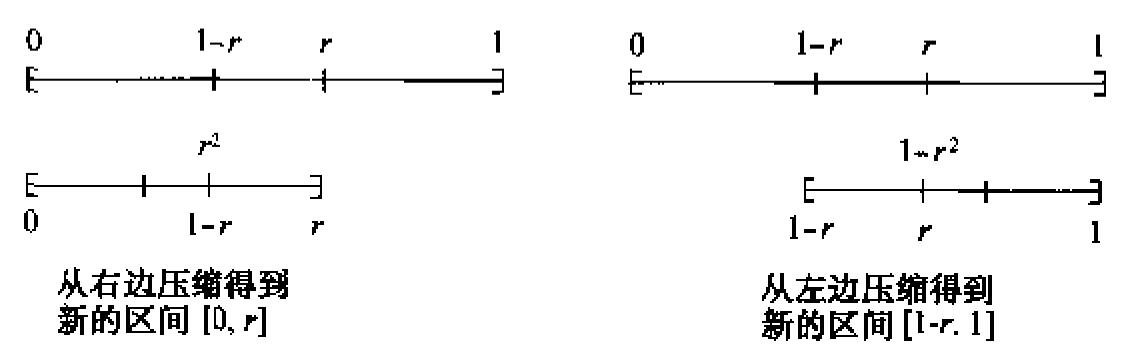


图 8.2　黄金分割搜索中的区间

定义 8.3(单峰函数)　设有函数 $f(x)$和区间 $I = [a, b]$,如果存在惟一的数 $p \in I$ 满足:

$f(x)$在区间$[a, p]$内递减　(1)

$f(x)$在区间$[p, b]$内递增　(2)

则称函数 $f(x)$为单峰函数。

如果 $f(x)$在区间$[a, b]$内为单峰函数,则可以用一个包含 $f(x)$最小值的子区间代替的区间。黄金分割搜索法要用两个内点 $c = a + (1 - r)(b - a)$和 $d = a + r(b - a)$,这里 r 是前面提到的黄金分割率。这些内点满足 $a < c < d < b$。$f(x)$为单峰函数的条件保证了函数值$f(c)$和 $f(d)$小于$\{f(a), f(b)\}$。下面要考虑两种情况(如图 8.3 所示):

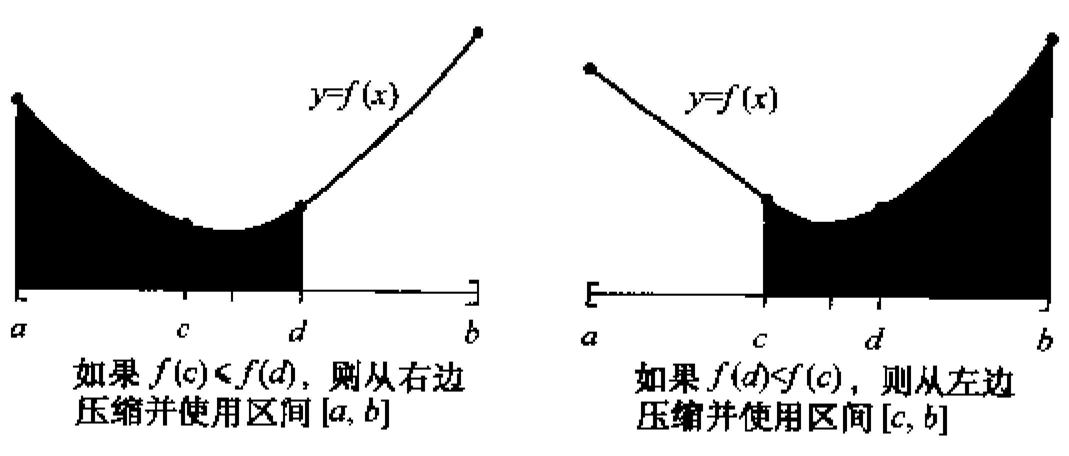


图 8.3　黄金分割搜索法的判定过程

如果 $f(c) \leqslant f(d)$,则最小值一定在子区间$[a, d]$中,这样可用 d 代替 b,并继续在新的子区间内搜索。如果 $f(d) < f(c)$,则最小值一定在子区间$[c, b]$,这样可用 c 代替 a,并继续在新的子区间内搜索。下面的例子比较了求根法和黄金分割搜索法。

例 8.2　求解单峰函数 $f(x) = x^2 - \sin(x)$在区间$[0,1]$内的最小值。

解:

通过求解 $f'(x) = 0$,求最小值。可用求根法来求函数导数式 $f'(x) = 2x - \cos(x)$等于零的解。由于 $f'(0) = -1$ 且 $f'(1) = 1.4596977$,所有 $f'(x)$的一个根位于区间$[0, 1]$内。初始值为 $p_0 = 0, p_1 = 1$,迭代过程如表 8.1 所示。

表 8.1　用割线法求解 $f'(x) = 2x - \cos(x) = 0$

k	p_k	$2p_k - \cos(p_k)$
0	0.0000000	-1.00000000
1	1.0000000	1.45969769
2	0.4065540	-0.10538092

(续表)

k	p_k	$2p_k - \cos(p_k)$
3	0.4465123	-0.00893398
4	0.4502137	0.00007329
5	0.4501836	-0.00000005

采用割线法得到结果是 $f'(0.4501836)=0$。二阶导数为 $f''(x)=2+\sin(x)$，通过计算可得 $f''(0.4501836)=2.435131>0$，因此，最小值为 $f(0.4501836)=-0.2324656$。

通过黄金分割搜索法求最小值。在每一步中，需要比较 $f(c)$ 和 $f(d)$ 的函数值，而且要决定是在区间 $[a,d]$ 内还是在 $[c,b]$ 内继续搜索。相关的计算如表 8.2 所示。

表 8.2 用黄金分割搜索法求解 $f(x)=x^2-\sin(x)$ 的最小值

k	a_k	c_k	d_k	b_k	$f(c_k)$	$f(d_k)$
0	0.0000000	0.3819660	0.6180340	1	-0.22684748	- -0.19746793
1	0.0000000	0.2360680	0.3819660	0.6180340	-0.17815339	0.22684748
2	0.2360680	0.3819660	0.4721360	0.6180340	-0.22684748	-0.23187724
3	0.3819660	0.4721360	0.5278640	0.6180340	-0.23187724	- -0.22504882
4	0.3819660	0.4376941	0.4721360	0.5278640	-0.23227594	-0.23187724
5	0.3819660	0.4164079	0.4376941	0.4721360	-0.23108238	-0.23227594
6	0.4164079	0.4376941	0.4508497	0.4721360	-0.23227594	-0.23246503
⋮	⋮	⋮	⋮	⋮	⋮	⋮
21	0.4501574	0.4501730	0.4501827	0.4501983	-0.23246558	-0.23246558
22	0.4501730	0.4501827	0.4501886	0.4501983	-0.23246558	-0.23246558
23	0.4501827	0.4501886	0.4501923	0.4501983	-0.23246558	-0.23246558

经过 23 次迭代后，区间缩小为 $[a_{23}, b_{23}]=[0.4501827, 0.4501983]$。这个区间的宽度为 0.0000156。然而，在这个区间两个端点的函数值小数点后有 8 位相同(即 $f(a_{23})\approx -0.23246558\approx f(b_{23})$)，因此算法终止。采用搜索法的一个问题是函数在最小值处可能比较平缓，使得精度受限。而使用割线法可得到更精确的结果 $p_5=0.4501836$。

尽管在本例中黄金分割搜索法的搜索速度较慢，但它可用于函数 $f(x)$ 不可导的情况。

8.1.2 求解 $f(x,y)$ 的极值

定义 8.1 可以很容易地扩展到多变量函数。设函数 $f(x,y)$ 在如下区域有定义：

$$R=\{(x,y):(x-p)^2+(y-q)^2<r^2\} \tag{3}$$

如果对每个点满足：

$$f(p,q)\leqslant f(x,y), \qquad (x,y)\in R \tag{4}$$

则函数 $f(x,y)$ 在点 (p,q) 有局部极小值。

如果对每个点满足：

$$f(x,y)\leqslant f(p,q), \qquad (x,y)\in R \tag{5}$$

则函数 $f(x,y)$ 在点 (p,q) 有局部极大值。

下面的极值的二阶导数测试是定理8.4的扩展。

定理8.5(二阶导数测试) 设有函数 $f(x,y)$,它的一阶偏导和二阶偏导在区间R内连续。设 $(p,q)\in R$ 是一个临界点,满足 $f_x(p,q)=0$ 且 $f_y(p,q)=0$。可用高阶偏导判定临界点的性质:

(i) 如果 $f_{xx}(p,q)f_{yy}(p,q)-f_{xy}^2(p,q)>0$ 且 $f_{xx}(p,q)>0$,则 $f(p,q)$ 是函数 $f(x,y)$ 的局部极小值。

(ii) 如果 $f_{xx}(p,q)f_{yy}(p,q)-f_{xy}^2(p,q)>0$ 且 $f_{xx}(p,q)<0$,则 $f(p,q)$ 是函数 $f(x,y)$ 的局部极大值。

(iii) 如果 $f_{xx}(p,q)f_{yy}(p,q)-f_{xy}^2(p,q)<0$,则函数 $f(x,y)$ 在点 (p,q) 没有局部极值。

(iv) 如果 $f_{xx}(p,q)f_{yy}(p,q)-f_{xy}^2(p,q)=0$,则测试不确定。

例8.3 求解函数 $f(x,y)=x^2-4x+y^2-y-xy$ 的最小值。

函数的一阶偏导为:

$$f_x(x,y)=2x-4-y \text{ 和 } f_y(x,y)=2y-1-x \tag{6}$$

使偏导为零,可得线性方程组:

$$\begin{aligned} 2x-y&=4 \\ -x+2y&=1 \end{aligned} \tag{7}$$

方程组(7)的解为 $(x,y)=(3,2)$。函数 $f(x,y)$ 的二阶偏导为:

$$f_{xx}(x,y)=2,f_{yy}(x,y)=2,f_{xy}(x,y)=-1$$

显见其满足定理8.5的情况(i),即:

$$f_{xx}(3,2)f_{yy}(3,2)-f_{xy}^2(3,2)=3>0 \text{ 且 } f_{xx}(3,2)=2>0$$

因此,函数 $f(x,y)$ 在点(3,2)处有局部极小值 $f(3,2)=-7$。

8.1.3 Nelder-Mead 法

Nelder和Mead提出了单纯形法,它可求解有多个变量的函数的局部极小值。对于两个变量,一个单纯形是三角形,而这个方法是一个模式搜索法,比较在三角形的3个顶点的函数值。最差的顶点即函数值 $f(x,y)$ 最大的顶点,被放弃并用一个新的顶点代替。然后形成一个新的三角形并继续进行搜索。这个过程生成一个三角形的序列(形状可能不同),满足在顶点的函数值越来越小。随着三角形的大小进一步减小,就可以找到极小值的坐标。

在算法中使用了名词-单纯形(在 N 维空间的广义三角形),并可求解有 N 个变量的函数极小值。它的计算有效且紧凑。

1. 初始三角形 *BGW*

假设要求解函数 $f(x,y)$ 的极小值。首先,给定一个三角形的3个顶点:$\boldsymbol{V}_k=(x_k,y_k)$,$k=1,2,3$。然后计算函数 $f(x,y)$ 在这3个点的值 $z_k=f(x_k,y_k)$,$k=1,2,3$。而且 $z_1\leqslant z_2\leqslant z_3$。使用下列符号表示:

$$\boldsymbol{B}=(x_1,y_1),\boldsymbol{G}=(x_2,y_2),\boldsymbol{W}=(x_3,y_3) \tag{8}$$

来帮助记忆 $\boldsymbol{B}$ 是最佳的顶点,$\boldsymbol{G}$ 是次最佳的顶点,而 $\boldsymbol{W}$ 是最差的顶点。

2. 良边(good side)的中点

构造过程中使用了连接 $\boldsymbol{B}$ 和 $\boldsymbol{G}$ 的线段的中点。它可通过计算坐标的平均值得到：

$$M = \frac{B + G}{2} = \left(\frac{x_1 + x_2}{2}, \frac{y_1 + y_2}{2}\right) \tag{9}$$

3. 反射点 R

当沿着三角形的边从 $\boldsymbol{W}$ 移到 $\boldsymbol{B}$ 时，函数值递减，而当沿着三角形的边从 $\boldsymbol{W}$ 移到 $\boldsymbol{G}$ 时，函数值也递减。因此以连接 $\boldsymbol{B}$ 和 $\boldsymbol{G}$ 的线为分界线，与 $\boldsymbol{W}$ 相对的点的函数值会更小。通过边 $\overline{BG}$ 对三角形进行反射可得到测试点 $\boldsymbol{R}$。为了求出 $\boldsymbol{R}$，首先求 $\overline{BG}$ 的中点 $\boldsymbol{M}$，然后画出经过 $\boldsymbol{W}$ 和 $\boldsymbol{M}$ 的线段，设它的长度为 d。从 $\boldsymbol{M}$ 点延伸线段，距离为 d，可到达顶点 $\boldsymbol{R}$ 的位置（如图 8.4 所示）。$\boldsymbol{R}$ 的向量公式为：

$$R = M + (M - W) = 2M - W \tag{10}$$

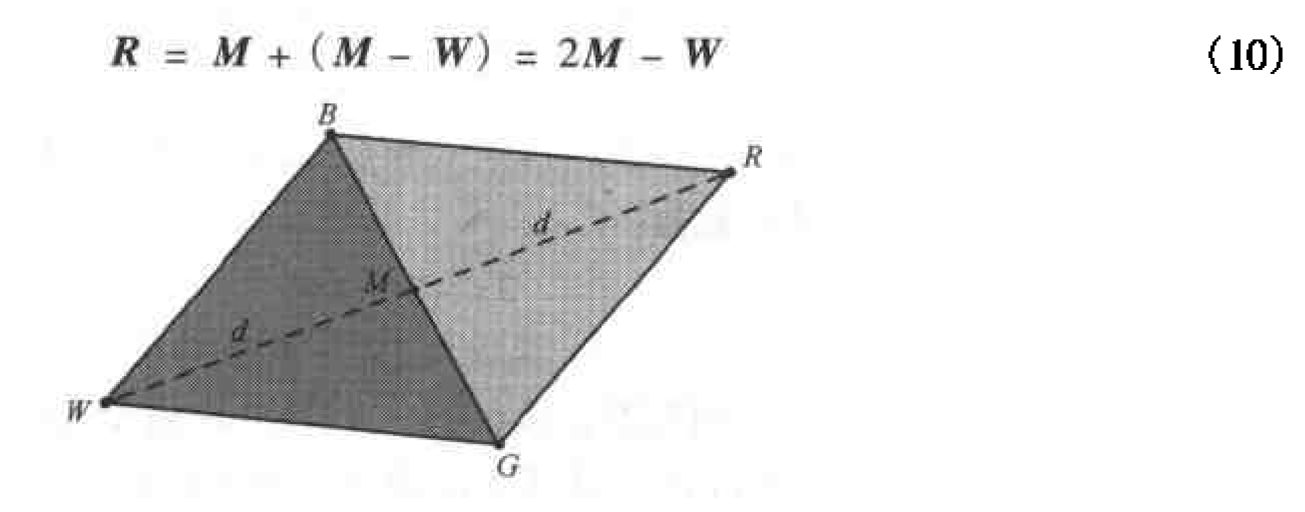


图 8.4 Nelder - Mead 法中的三角形△*BGW*，中点 $\boldsymbol{M}$ 和反射点 $\boldsymbol{R}$

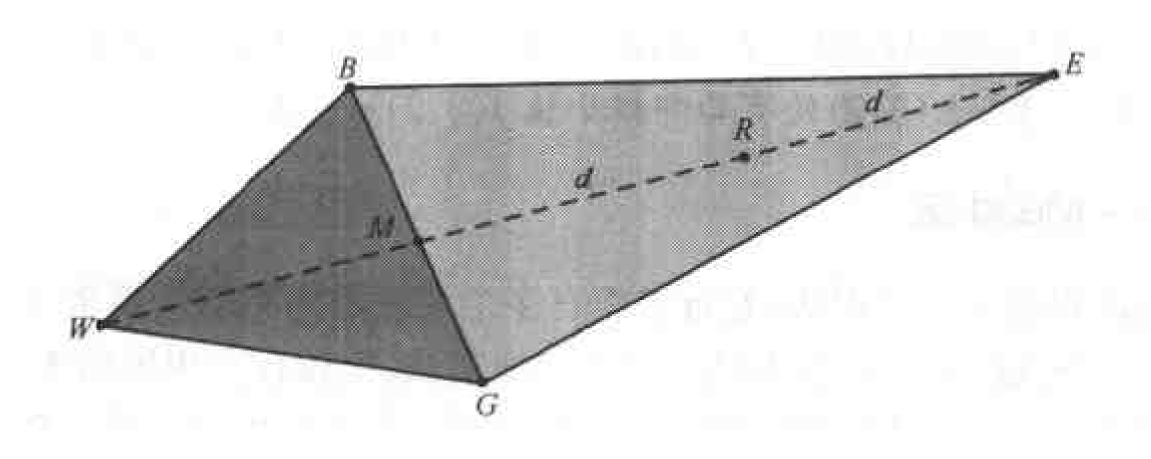


图 8.5 三角形△*BGW*，点 *R* 和开拓点 *E*

2. 开拓点 *E*

如果位于 $\boldsymbol{R}$ 处的函数值小于位于 $\boldsymbol{W}$ 处的函数值，则求解极小值的方向是正确的。可能极小值只比 $\boldsymbol{R}$ 远一点。因此可进一步将从 $\boldsymbol{M}$ 到 $\boldsymbol{R}$ 的线段延伸到 $\boldsymbol{E}$。其形状为扩展三角形 $\boldsymbol{BGE}$。点 $\boldsymbol{E}$ 位于沿 $\boldsymbol{M}$ 到 $\boldsymbol{R}$ 的线段方向，从 $\boldsymbol{R}$ 延伸到距离为 d 的位置。如果位于 $\boldsymbol{E}$ 处的函数值小于位于 $\boldsymbol{R}$ 处的函数值，则找到一个比 $\boldsymbol{R}$ 更好的顶点。$\boldsymbol{E}$ 的向量公式为：

$$E = R + (R - M) = 2R - M \tag{11}$$

3. 收缩点 *C*

如果位于 $\boldsymbol{R}$ 和 $\boldsymbol{W}$ 处的函数值相同，则需要测试另一个点。可能在 $\boldsymbol{M}$ 处的函数值更小，

但不能用 $\boldsymbol{M}$ 代替 $\boldsymbol{W}$，因为必须满足三角形的条件。分别考虑线段 $\overline{WM}$ 和 $\overline{MR}$ 的中点 $\boldsymbol{C}_1$ 和 $\boldsymbol{C}_2$（如图 8.6 所示），将具有较小函数值的点称为 $\boldsymbol{C}$，并形成新的三角形 $\boldsymbol{BGC}$。注释：在二维情况下对 $\boldsymbol{C}_1$ 和 $\boldsymbol{C}_2$ 的选择看起来不适当，但这对于高维情况非常重要。

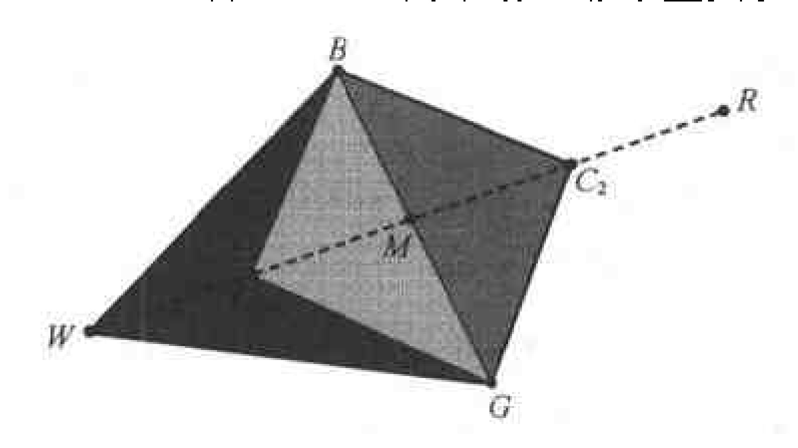


图 8.6　Nelder – Mead 法的收缩点 $\boldsymbol{C}_1$ 和 $\boldsymbol{C}_2$

4. 沿 B 方向压缩

如果位于 $\boldsymbol{C}$ 处的函数值不小于位于 $\boldsymbol{W}$ 处的函数值，则点 $\boldsymbol{G}$ 和 $\boldsymbol{W}$ 必须沿 $\boldsymbol{B}$ 方向压缩（如图 8.7 所示）。用 $\boldsymbol{M}$ 代替 $\boldsymbol{G}$，并用 $\boldsymbol{S}$ 代替 $\boldsymbol{W}$，$\boldsymbol{M}$ 是 $\boldsymbol{B}$ 和 $\boldsymbol{G}$ 之间线段的中点，$\boldsymbol{S}$ 是 $\boldsymbol{B}$ 和 $\boldsymbol{W}$ 之间线段的中点。

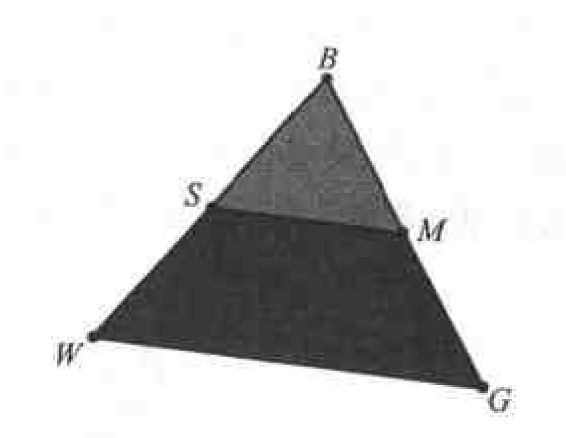
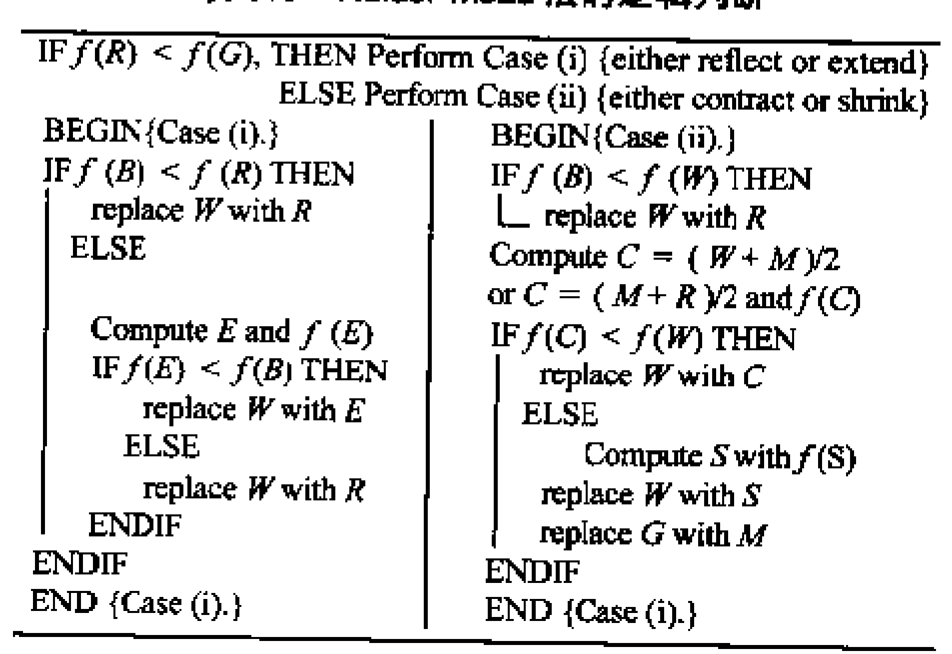


图 8.7　沿 $\boldsymbol{B}$ 方向压缩三角形

5. 每一步的逻辑判断

一个高效的算法应该只进行必要的计算。在每一步中，要求出一个新的顶点以代替 $\boldsymbol{W}$。一旦找到这个顶点，就终止这一步。二维情况下的逻辑判断细节如表 8.3 所示。

表 8.3　Nelder-Mead 法的逻辑判断



```
IF f(R) < f(G), THEN Perform Case (i) {either reflect or extend}
                ELSE Perform Case (ii) {either contract or shrink}
```

Case (i)	Case (ii)
BEGIN{Case (i).}	BEGIN{Case (ii).}
IF f (B) < f (R) THEN	IF f (B) < f (W) THEN
replace W with R	replace W with R
ELSE	Compute C = (W + M)/2
	or C = (M + R)/2 and f(C)
Compute E and f (E)	IF f(C) < f(W) THEN
IF f(E) < f(B) THEN	replace W with C
replace W with E	ELSE
ELSE	Compute S with f(S)
replace W with R	replace W with S
ENDIF	replace G with M
ENDIF	ENDIF
END {Case (i).}	END {Case (i).}

例 8.4 用 Nelder-Mead 算法求 $f(x,y)=x^2-4x+y^2-y-xy$ 的极小值。初始点为：

$$V_1=(0,0),\quad V_2=(1.2,0.0),\quad V_3=(0.0,0.8)$$

在这些点处的函数值为：

$$f(0,0)=0.0,\quad f(1.2,0.0)=-3.36,\quad f(0.0,0.8)=-0.16$$

比较这些函数值以得到 $\boldsymbol{B},\boldsymbol{G},\boldsymbol{W}$：

$$\boldsymbol{B}=(1.2,0.0),\boldsymbol{G}=(0.0,0.8),\boldsymbol{W}=(0,0)$$

顶点 $\boldsymbol{W}=(0,0)$将被代替。点 $\boldsymbol{M}$ 和 $\boldsymbol{R}$ 为：

$$\boldsymbol{M}=\frac{\boldsymbol{B}+\boldsymbol{G}}{2}=(0.6,0.4) \text{ 和 } \boldsymbol{R}=2\boldsymbol{M}-\boldsymbol{W}=(1.2,0.8)$$

函数值 $f(\boldsymbol{R})=f(1.2,0.8)=-4.48$ 小于 $f(\boldsymbol{G})$，所以符合情况(i)。由于 $f(\boldsymbol{R})\leqslant f(\boldsymbol{B})$，所以移动方向正确，并构造顶点 $\boldsymbol{E}$ 为：

$$\boldsymbol{E}=2\boldsymbol{R}-\boldsymbol{M}=2(1.2,0.8)-(0.6,0.4)=(1.8,1.2)$$

函数值 $f(\boldsymbol{E})=f(1.8,1.2)=-5.88$ 小于 $f(\boldsymbol{B})$，因此新的三角形顶点为：

$$\boldsymbol{V}_1=(1.8,1.2),\boldsymbol{V}_2=(1.2,0.0),\boldsymbol{V}_3=(0.0,0.8)$$

继续这个过程并生成一个三角形序列，最终收敛到解(3,2)(如图 8.8 所示)。表 8.4 给出了每一步中这些顶点的函数值。计算机实现的算法执行了 33 步，得到最佳顶点为 $\boldsymbol{B}=(2.99996456,1.99983839)$，函数值为 $f(\boldsymbol{B})=-6.99999998$。这些值是例 8.3 中 $f(3,2)=-7$ 的近似值。迭代在到达点(3,2)之前停止的原因是函数在极小值附近比较平缓。经过检查，函数值 $f(\boldsymbol{B}),f(\boldsymbol{G}),f(\boldsymbol{W})$是相同的(这是一个舍入误差的例子)，同时算法终止。

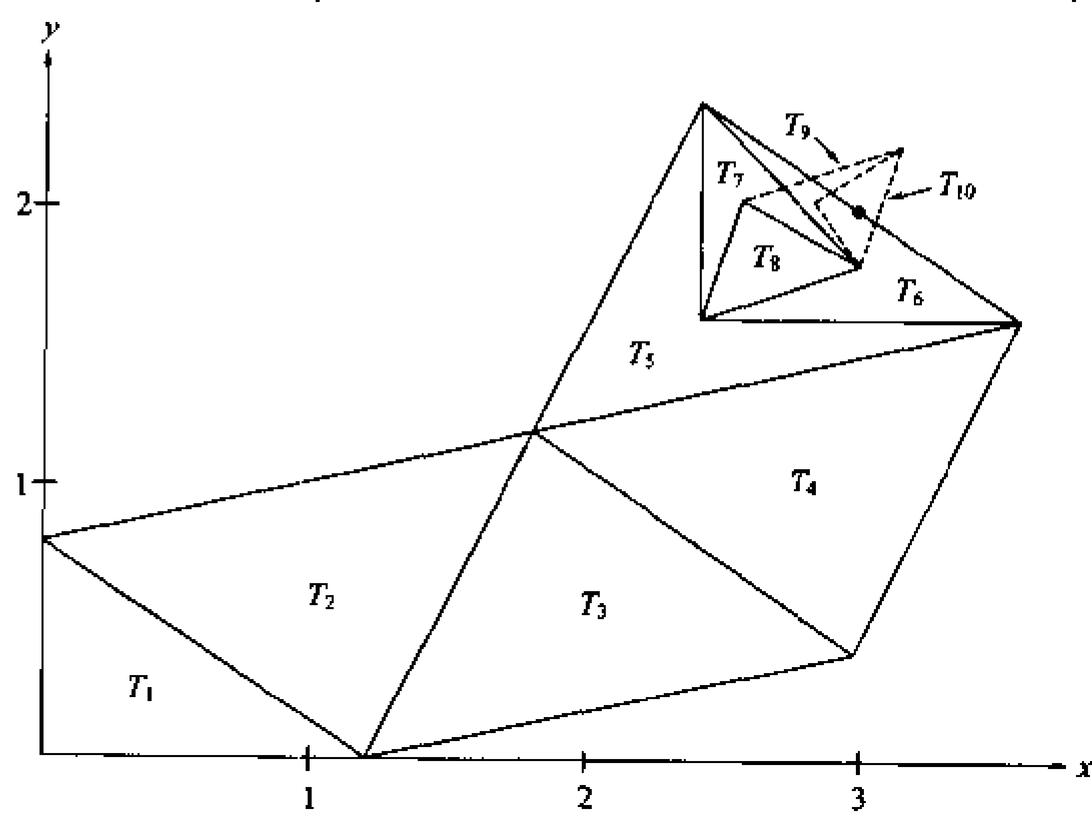


图 8.8 Nelder-Mead 法中三角形$\{T_k\}$序列收敛到点(3,2)

8.1.4 根据导数求极小值

设 $f(x)$在区间$[a,b]$内是单峰函数，并在 $x=p$ 处有惟一的极小值，导数 $f'(x)$在位于区间(a,b)内的所有点上有定义。设初始点 p_0 位于区间(a,b)内。如果 $f'(p_0)<0$，则极小值的点 p 位于p_0 的右边。如果 $f'(p_0)>0$，则 p 位于p_0 的左边(如图 8.9 所示)。

表8.4 例8.4中不同三角形顶点的函数值

k	最佳点	较好点	最差点
1	$f(1.2,0.0)=-3.36$	$f(0.0,0.8)=-0.16$	$f(0.0,0.0)=0.00$
2	$f(1.8,1.2)=-5.88$	$f(1.2,0.0)=-3.36$	$f(0.0,0.8)=-0.16$
3	$f(1.8,1.2)=-5.88$	$f(3.0,0.4)=-4.44$	$f(1.2,0.0)=-3.36$
4	$f(3.6,1.6)=-6.24$	$f(1.8,1.2)=-5.88$	$f(3.0,0.4)=-4.44$
5	$f(3.6,1.6)=-6.24$	$f(2.4,2.4)=-6.24$	$f(1.8,1.2)=-5.88$
6	$f(2.4,1.6)=-6.72$	$f(3.6,1.6)=-6.24$	$f(2.4,2.4)=-6.24$
7	$f(3.0,1.8)=-6.96$	$f(2.4,1.6)=-6.72$	$f(2.4,2.4)=-6.24$
8	$f(3.0,1.8)=-6.96$	$f(2.55,2.05)=-6.7725$	$f(2.4,1.6)=-6.72$
9	$f(3.0,1.8)=-6.96$	$f(3.15,2.25)=-6.9525$	$f(2.55,2.05)=-6.7725$
10	$f(3.0,1.8)=-6.96$	$f(2.8125,2.0375)=-6.95640625$	$f(3.15,2.25)=-6.9525$

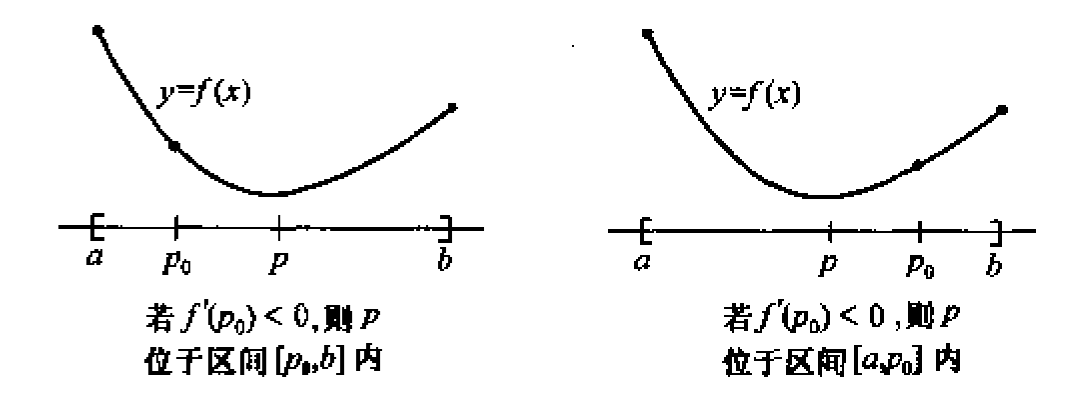


图8.9 用$f'(x)$求解位于区间$[a,b]$内的单峰函数$f(x)$的极小值

1. 求极小值的测试值

第一步是得到3个测试值：

$$p_0,\quad p_1=p_0+h,\quad p_2=p_0+2h \tag{12}$$

使得：

$$f(p_0)>f(p_1)\text{ 且 }f(p_1)<f(p_2) \tag{13}$$

设$f'(p_0)<0$,则$p_0<p$且步长h须为正数。容易找到h使得式(12)中的3个点满足式(13)。在式(12)中,设初始值$h=1$(前提为$a+1<b$),如果前提不满足,则设$h=1/2$,以此推类。

情况(i)：如果式(13)得到满足,则结束。

情况(ii)：如果$f(p_0)>f(p_1)$且$f(p_1)>f(p_2)$,则$p_2<p$。需要检查更靠右的点。步长乘以2,并重复执行。

情况(iii)：如果$f(p_0)\leqslant f(p_1)$,则跳过了p且h太大。需要检查更接近p_0的点。将步长乘以$\frac{1}{2}$,并重复执行。

当$f'(p_0)>0$时,步长h须为负数,处理过程类似上面的(i)到(iii)。

2. 求极小值p的二次逼近

最后,可得到满足式(13)的式(12)中的3个点。可通过二次插值法求解p_{min},它是p的近

［Ｇｅｎｅｒａｌ Ｉｎｆｏｒｍａｔｉｏｎ］
书名＝数值方法 （ＭＡＴＬＡＢ版） （第三版）
作者＝
页数＝３０５
ＳＳ号＝０
出版日期＝
Ｖｓｓ号＝７６０５９４０１

目录

目　　录

第１章预备知识
１．１微积分回顾
１．１．１极限和连续性
１．１．２微分函数
１．１．３积分
１．１．４级数
１．１．５多项式求值
１．１．６微积分回顾的练习题
１．２二进制数
１．２．１二进制数
１．２．２序列与级数
１．２．３二进制分数
１．２．４二进制移位
１．２．５科学计数法
１．２．６机器数
１．２．７计算机精度
１．２．８计算机浮点数
１．２．９二进制数的练习
１．３误差分析
１．３．１截断误差
１．３．２舍入误差
１．３．３舍去和舍入
１．３．４精度损失
１．３．５　Ｏ（ｈｎ）阶逼近
１．３．６序列的收敛阶
１．３．７误差传播
１．３．８数据的不确定性
１．３．９误差分析的练习
１．３．１０算法和程序
第２章非线性方程ｆ（ｘ）＝０的解法
２．１求解ｘ＝ｇ（ｘ）的迭代法
２．１．１寻求固定点
２．１．２固定点迭代的图形解释
２．１．３绝对误差和相对误差
２．１．４求解ｘ＝ｇ（ｘ）迭代过程的练习
２．１．５算法和程序
２．２定位一个根的划分方法（ｂｍｃｋｅｔｉｎｇ　ｍｅｔｈｏｄｓ）
２．２．１波尔察诺（Ｂｏｌｚａｎｏ）二分法
２．２．２试值法的收敛性
２．２．３划分方法练习
２．２．４算法和程序
２．３初始近似值和收敛判定准则
２．３．１检测收敛性
２．３．２有问题的函数（ＴｒｏｕｂｌｅＳｏｍｅ　Ｆｕｎｃｔｉｏｎｓ）
２．３．３初始近似值的练习
２．３．４算法和程序
２．４牛顿拉夫申（Ｎｅｗｔｏｎ－Ｒａｐｈｓｏｎ）法和割线法
２．４．１求根的斜率法
２．４．２被零除错误
２．４．３收敛速度
２．４．４缺陷
２．４．５割线法
２．４．６加速收敛
２．４．７牛顿拉夫申法和割线法的练习
２．４．８算法和程序
２．５　Ａｉｔｋｅｎ过程、Ｓｔｅｆｆｅｎｓｅｎ法和Ｍｕｌｌｅｒ法（可选）
２．５．１　Ａｉｔｋｅｎ过程
２．５．２　Ｍｕｌｌｅｒ法

２．５．３方法之间的比较
２．５．４　Ａｉｔｋｅｎ法、Ｓｔｅｆｆｅｎｓｅｎ法和Ｍｕｌｌｅｒ法的练习
２．５．５算法和程序
第３章线性方程组ＡＸ＝Ｂ的数值解法
３．１向量和矩阵介绍
３．１．１矩阵和二维数组
３．１．２向量和矩阵简介的练习
３．２向量和矩阵的性质
３．２．１矩阵乘
３．２．２特殊矩阵
３．２．３非奇异矩阵的逆
３．２．４行列式
３．２．５平面旋转
３２６　ＭＡＴＬＡＢ
３．２．７向量和矩阵性质的练习
３．２．８算法和程序
３．３上三角线性方程组
３．３．１上三角线性方程组的练习
３．３．２算法和程序
３．４高斯消去法和选主元
３．４．１选主元以避免ａｐｐ（ｐ）＝０
３．４．２选主元以减少误差
３．４．３病态情况
３．４．４　ＭＡＴＬＡＢ
３．４．５高斯消去法和选主元的练习
３．４．６算法和程序
３．５三角分解法
３．５．１线性方程组的解
３．５．２三角分解法
３．５．３计算复杂性
３．５．４置换矩阵
３．５．５扩展高斯消去过程
３．５．６　ＭＡＴＬＡＢ
３．５．７三角分解法的练习
３．５．８算法和程序
３．６求解线性方程组的迭代法
３．６．１雅克比迭代
３．６．２　Ｇａｕｓｓ－Ｓｅｉｄｅｌ迭代法
３．６．３收敛性
３．６．４求解线性方程组的迭代法的练习
３．６．５算法和程序
３．７非线性方程组的迭代法：Ｓｅｉｄｅｌ法和牛顿法（可选）
３．７．１理论
３．７．２广义微分
３．７．３接近固定点处的收敛性
３．７．４　Ｓｅｉｄｅｌ迭代
３．７．５求解非线性方程组的牛顿法
３．７．６牛顿法概要
３７７　ＭＡＴＬＡＢ
３．７．８求解非线性方程组的迭代法的练习
３．７．９算法和程序
第４章　　插值与多项式逼近
４．１泰勒级数和函数计算
４．１．１多项式计算方法
４．１．２习题
４．１．３算法与程序
４．２插值介绍
４．２．１习题
４．２．２算法与程序
４．３拉格朗日逼近

４．３．１误差项和误差界
４．３．２比较精度与Ｏ（ｈｎ＋１）
４．３．３　ＭＡＴＬＡＢ
４．３．４习题
４．３．５算法与程序
４．４牛顿多项式
４．４．１嵌套乘法
４．４．２多项式逼近、节点及中心
４．４．３习题
４．４．４算法与程序
４．５切比雪夫多项式（可选）
４．５．１切比雪夫多项式性质
４．５．２最小上界
４．５．３等距节点
４．５．４切比雪夫节点
４．５．５龙格现象
４．５．６区间变换
４．５．７正交性质
４．５．８　ＭＡＴＬＡＢ
４．５．９习题
４．５．１０算法与程序
４．６帕德逼近
４．６．１连分式
４．６．２习题
４．６．３算法与程序
第５章曲线拟合
５．１最小二乘拟合曲线
５．１．１求最小二乘曲线
５．１．２幂函数拟合ｙ＝Ａｘｍ
５．１．３最小二乘拟合曲线的练习
５．１．４算法和程序
５．２曲线拟合
５．２．１对ｙ＝ＣｅＡｘ线性化方法
５．２．２求解ｙ＝ＣｅＡｘ的非线性最小二乘法
５．２．３数据线性化变换
５．２．４线性最小二乘法
５．２．５矩阵公式
５．２．６多项式拟合
５．２．７多项式摆动
５．２．８曲线拟合的练习
５．２．９算法和程序
５．３样条函数插值
５．３．１分段线性插值
５．３．２分段三次样条曲线
５．３．３三次样条的存在性
５．３．４构造三次样条
５．３．５端点约束
５．３．６三次样条曲线的适宜性
５．３．７样条函数插值的练习
５．３．８算法和程序
５．４傅里叶级数和三角多项式
５．４．１三角多项式逼近
５．４．２傅里叶级数和三角多项式的练习
５．４．３算法和程序
第６章数值微分
６．１导数的近似值
６．１．１差商的极限
６．１．２　中心差分公式
６．１．３误差分析和优化步长
６．１．４　Ｒｉｃｈａｒｄｓｏｎ外推法

６．１．５导数近似值的练习
６．１．６算法和程序
６．２数值差分公式
６．２．１更多的中心差分公式
６．２．２误差分析
６．２．３拉格朗日多项式微分
６．２．４牛顿多项式微分
６．２．５数值微分公式的练习
６．２．６算法和程序
第７章数值积分
７．１积分简介
７．１．１习题
７．２组合梯形公式和辛普生公式
７．２．１误差分析
７．２．２习题
７．２．３算法与程序
７．３递归公式与龙贝格积分
７．３．１龙贝格积分
７．３．２习题
７．３．３算法与程序
７．４自适应积分
７．４．１区间细分（ｒｅｆｉｎｅｍｅｎｔ）
７．４．２精度测试
７．４．３算法与程序
７．５高斯－勒让德积分（可选）
７．５．１习题
７．５．２算法与程序
第８章数值优化
８．１函数极小值
８．１．１搜索方法
８．１．２求解ｆ（ｘ，ｙ）的极值
８．１．３　Ｎｅｌｄｅｒ－Ｍｅａｄ法
８．１．４根据导数求极小值
８．１．５最速下降法
８．１．６求解函数极小值的练习
８．１．７算法和程序
第９章微分方程求解
９．１微分方程导论
９．１．１初值问题
９．１．２几何解释
９．１．３习题
９．２欧拉方法
９．２．１几何描述
９．２．２步长与误差
９．２．３习题
９．２．４算法与程序
９．３休恩方法
９．３．１步长与误差
９．３．２习题
９．３．３算法与程序
９．４泰勒级数法
９．４．１习题
９．４．２算法与程序
９．５龙格－库塔方法
９．５．１关于该方法的讨论
９．５．２步长与误差
９．５．３　Ｎ＝２的龙格－库塔方法
９．５．４龙格－库塔－费尔博格方法（ＲＫＦ４５）
９．５．５习题
９．５．６算法与程序

９．６预测－校正方法
９．６．１阿达姆斯－巴什弗斯－摩尔顿方法
９．６．２误差估计与校正
９．６．３实际考虑
９．６．４米尔尼－辛普生方法
９．６．５误差估计与校正
９．６．６　正确的步长
９．６．７习题
９．６．８程序与算法
９．７微分方程组
９．７．１数值解
９．７．２　　高阶微分方程
９．７．３习题
９．７．４算法与程序
９．８边值问题
９．８．１分解为两个初值问题：线性打靶法
９．８．２习题
９．８．３算法与程序
９．９有限差分方法
９．９．１习题
９．９．２算法与程序
第１０章偏微分方程数值解
１０．１双曲型方程
１０．１．１波动方程
１０．１．２差分方程
１０．１．３初始值
１０．１．４　Ｄ＇Ａｌｅｍｂｅｒｔ方法
１０．１．５给定的两个确定行
１０．１．６双曲线型方程的练习
１０．１．７算法和程序
１０．２抛物型方程
１０．２．１热传导方程
１０．２．２差分方程
１０．２．３　Ｃｒａｎｋ－Ｎｉｃｈｏｌｓｏｎ法
１０．２．４抛物型方程的练习
１０．２．５算法和程序
１０．３椭圆型方程
１０．３．１　Ｌａｐｌａｃｅ差分方程
１０．３．２建立线性方程组
１０．３．３导数边界条件
１０．３．４迭代方法
１０．３．５　Ｐｏｉｓｓｏｎ方程和Ｈｅｌｍｈｏｌｔｚ方程
１０．３．６改进
１０．３．７椭圆型方程的练习
１０．３．８算法和程序
第１１章特征值与特征向量
１１．１齐次方程组：特征值问题
１１．１．１背景知识
１１．１．２特征值
１１．１．３对角化
１１．１．４对称性的优势
１１．１．５特征值范围估计
１１．１．６方法综述
１１．１．７齐次方程组：特征值问题的练习
１１．２幂方法
１１．２．１收敛速度
１１．２．２移位反幂法
１１．２．３幂法的练习
１１．２．４算法和程序
１１．３雅克比方法

１１．３．１平面旋转变换
１１．３．２相似和正交变换
１１．３．３雅克比序列的变换
１１．３．４一般步骤
１１．３．５使ｄｐｑ和ｄｑｐ为零
１１．３．６一般步骤总结
１１．３．７修正矩阵的特征值
１１．３．８消去ａｐｑ的策略
１１．３．９雅克比法的练习
１１．３．１０算法和程序
１１．４对称矩阵的特征值
１１．４．１　Ｈｏｕｓｅｈｏｌｄｅｒ法
１１．４．２　Ｈｏｕｓｅｈｏｌｄｅｒ变换
１１．４．３三对角形式归约
１１．４．４ＱＲ法
１１．４．５加速移位
附录ＭＡＴＬＡＢ介绍
参考文献
习题答案